Dreaming up scale invariance via inverse renormalization group

Adam Rançon,[1, *] Ulysse Rançon,[2, †] Tomislav Ivek,[3, ‡] and Ivan Balog[3, §]

[1] *Univ. Lille, CNRS, UMR 8523 – PhLAM – Laboratoire de Physique des Lasers Atomes et Molécules, F-59000 Lille, France*
[2] *CerCo UMR 5549, CNRS – Université Toulouse III, Toulouse, France*
[3] *Institute of Physics, Bijenička cesta 46, HR-10001 Zagreb, Croatia*

We explore how minimal neural networks can invert the renormalization group (RG) coarse-graining procedure in the two-dimensional Ising model, effectively "dreaming up" microscopic configurations from coarse-grained states. This task—formally impossible at the level of configurations—can be approached probabilistically, allowing machine learning models to reconstruct scale-invariant distributions without relying on microscopic input. We demonstrate that even neural networks with as few as three trainable parameters can learn to generate critical configurations, reproducing the scaling behavior of observables such as magnetic susceptibility, heat capacity, and Binder ratios. A real-space renormalization group analysis of the generated configurations confirms that the models capture not only scale invariance but also reproduce nontrivial eigenvalues of the RG transformation. Surprisingly, we find that increasing network complexity by introducing multiple layers offers no significant benefit. These findings suggest that simple local rules, akin to those generating fractal structures, are sufficient to encode the universality of critical phenomena, opening the door to efficient generative models of statistical ensembles in physics.

I. INTRODUCTION

Universal behavior near critical points is a cornerstone of statistical physics. In systems undergoing second-order phase transitions, observables such as correlation functions and susceptibilities exhibit power-law scaling that is largely independent of microscopic details. The renormalization group (RG) framework elegantly explains this universality by describing how short-range fluctuations are systematically averaged out to reveal long-range scale-invariant properties [1]. Yet, this very process of coarse-graining is inherently lossy and one-way: RG eliminates information about microscopic degrees of freedom and cannot, in general, be inverted. However, it bears a striking resemblance to various techniques in Deep Learning (DL) that also focus on capturing essential features while discarding superfluous details. This connection has been under intensive scrutiny recently [2–9].

Machine learning (ML) and DL have become foundational tools across modern science and engineering. These methods have dramatically improved the state-of-the-art in tasks ranging from image and speech recognition to drug discovery and genomics, and they have entered virtually every scientific discipline as indispensable techniques for data analysis and modeling [10]. Over the last decade, deep learning has emerged as a powerful tool for data transformation, particularly in the domain of image processing where super-resolution plays a central role in reconstructing fine-scale structures from coarse data [11, 12]. Convolutional neural networks and their generative extensions have demonstrated remarkable success in upscaling complex data structures, both in
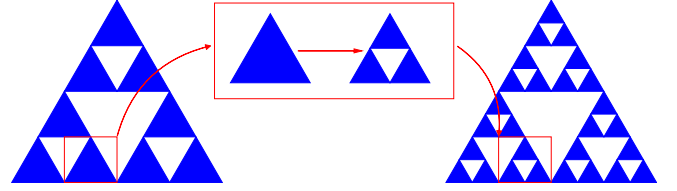

FIG. 1. Sierpiński triangle is a self-similar structure (a fractal) generated by an iterative application of a simple local rule (inset). A critical configuration of spins constitutes a fractal object as well, which leads us to explore the concept of using simple local neural networks for dreaming up critical spin configurations.

general imaging applications and in contexts where outputs must respect physical constraints [13, 14]. In particular, generative models such as Generative Adversarial Networks (GANs) [15], Variational Autoencoders (VAEs) [16], and, more recently, diffusion models [17], have significantly advanced the ability to synthesize high-resolution, high-dimensional data from low-dimensional inputs by learning the underlying data distributions [18–20]. These models enable the inference of plausible fine-grained details, yielding high-quality reconstructions that are both diverse and physically meaningful.

This capability is naturally aligned with the challenge of inverting renormalization group (RG) transformations, which has been the focus of recent works [21–25], demonstrating that complex neural networks trained on small system configurations can generate configurations for larger systems that qualitatively—and sometimes quantitatively—reproduce critical behavior. At the same time, phase-characterization models have evolved from early supervised and unsupervised architectures, using techniques like principal component analysis and autoencoders to detect transitions and learn emergent order

* adam.rancon@univ-lille.fr
† ulysse.rancon@cnrs.fr
‡ tivek@ifs.hr
§ balog@ifs.hr

parameters directly from raw Ising spin data [26, 27], to increasingly complex structures such as graph neural networks that encode interaction topology [28], attention-based transformers that capture long-range correlations [29], and symmetry-equivariant architectures enforcing lattice and spin-flip invariances [30]. While such approaches often greatly increase model complexity, it remains unclear if they are useful for inverting RG transformations or indeed what the minimal conditions are for a neural network to achieve scale invariance.

In this work, we revisit this problem from a minimalistic perspective. Rather than employing deep or highly parameterized models, we ask: *What is the simplest neural network that can learn to invert RG and reproduce scale-invariant physics?* By training single-layer convolutional networks with as few as three parameters, we show that it is possible to generate critical Ising configurations that reproduce key thermodynamic observables and universal finite-size scaling relations. Remarkably, we find that such simple models can match and in some respects outperform more complex architectures. The resulting configurations obey universal scaling laws and display stable RG eigenvalue spectra, confirming that even minimal models can learn the essential features of criticality. That a very simple network, with straightforward transformation rules, is able to produce scale invariance might not be too surprising in hindsight. Indeed, this is reminiscent of constructing fractals by local rules, such as e.g. the Sierpiński triangle, where simple iterative procedures yield complex, self-similar structures such as illustrated in Fig. 1.

Here, we focus on the Ising model, the standard model of statistical physics, known for its simplicity and rich critical behavior that makes it a classic example of universality in phase transitions. The model describes a two-dimensional lattice of spins, which interact via a Hamiltonian

$$H(\boldsymbol{\sigma}) = -K \sum_{\langle i,j \rangle} \sigma_i \sigma_j, \tag{1}$$

with ferromagnetic interaction K (proportional to the inverse temperature), and the sum is over nearest neighbors. This model undergoes a phase transition at a critical coupling, K_c, separating the paramagnetic phase, where spins are weakly correlated, from the ferromagnetic phase, where spins align. Furthermore, the physics close to the transition exhibits a universal behavior characterized by divergences in quantities such as magnetic susceptibility and heat capacity. These quantities follow power-law scaling with critical exponents that are independent of the system's microscopic details—a hallmark of universality. Despite its simplicity, the Ising model's critical properties are applicable to a broad class of physical systems, making it a paradigmatic example of a universality class in statistical physics.

Our objective in this study is to determine whether simple neural networks, with minimal parameters, can capture the scale-invariant properties of the Ising model

at criticality. We explore whether such networks can "dream up" configurations that reflect the universal characteristics of critical systems. Inspired by fractal patterns, which arise from simple, iterative rules, we focus on a minimalistic network architecture with linear filters of varying kernel sizes, assessing their capacity to learn universal scaling.

To validate our approach, we perform a finite-size scaling analysis, examining how well these networks reproduce known critical behavior. For a more stringent analysis, we conduct a Real Space Renormalization Group (RSRG) analysis on upscaled configurations to test the networks' learning of scale invariance. Our surprising results show that even these simple networks can approximate universal features of criticality effectively, often outperforming more complex models.

The manuscript is organized as follows. Section II describes the concepts of downscaling and upscaling and their connection to the renormalization group and scale invariance. In Sec. III, we introduce the simple neural network we study, its training and the upscaling procedure. In Sec. IV, we demonstrate the ability of this net to capture scale invariance by studying finite-size scaling of various observables, while in Sec. V we show that it also captures the relevant properties of the renormalization group. Finally, we discuss in Sec. VI how more complex models do not perform better to these tasks. We give our conclusions in Sec. VII.

The code used for model training and evaluation is available at this GitHub repository to allow reproducibility.

II. DOWNSCALING, UPSCALING, AND SCALE INVARIANCE

The probability distribution of a generic Ising model of linear size L for a given spin configuration $\boldsymbol{\sigma} = \{\sigma_i\}_{i=1,\cdots,L^2}$, $\sigma_i = \pm 1$, can be written as

$$P(\boldsymbol{\sigma}) = e^{-H(\boldsymbol{\sigma})}, \tag{2}$$

in terms of the generic Hamiltonian (the temperature is factored in the Hamiltonian)

$$H(\boldsymbol{\sigma}) = \sum_{\alpha} K_{\alpha} S_{\alpha}(\boldsymbol{\sigma}). \tag{3}$$

Here S_{α} are local spin operators (for instance nearest neighbor interaction) and K_{α} are the corresponding coupling constants. The normalization of P is given by K_0 (with $S_0(\boldsymbol{\sigma}) = 1$).

Kadanoff's blocking procedure from a spin configuration $\boldsymbol{\sigma}$ to a (coarse-grained) block-spin configuration $\boldsymbol{\mu}$ can be reframed in a probabilistic setting by introducing the conditional probability $D(\boldsymbol{\mu}|\boldsymbol{\sigma})$ and a joint probability $P(\boldsymbol{\sigma}, \boldsymbol{\mu}) = D(\boldsymbol{\mu}|\boldsymbol{\sigma})P(\boldsymbol{\sigma})$. Assuming the block-spins are constructed such that the linear size of the system is

halved, one constructs the probability distribution of the block-spins as

$$P_1(\boldsymbol{\mu}) = \sum_{\boldsymbol{\sigma}} P(\boldsymbol{\sigma}, \boldsymbol{\mu}) = \sum_{\boldsymbol{\sigma}} D(\boldsymbol{\mu}|\boldsymbol{\sigma}) P(\boldsymbol{\sigma}). \qquad (4)$$

The block-spinning kernel D is chosen such that $\sum_{\boldsymbol{\mu}} D(\boldsymbol{\mu}|\boldsymbol{\sigma}) = 1$ satisfies the so-called trace condition, $\sum_{\boldsymbol{\mu}} P(\boldsymbol{\sigma}, \boldsymbol{\mu}) = P(\boldsymbol{\sigma})$, to leave the thermodynamics of the system invariant. This transformation can be deterministic or probabilistic but is not invertible: it should allow for forgetting irrelevant (microscopic) information. In the following, we will consider the standard majority rule to block-spin, i.e. the lattice is divided into 2×2 blocks where each block is transformed to a spin with the same state as the majority of spins in the block. In case of ambiguity, we randomly pick the block-spin to be ± 1 with probability $1/2$. The blocking procedure can also be interpreted in probabilistic and information theoretical setups in terms of Restricted Boltzmann Machine or more complicated neural nets and optimized upon [5, 6, 31–34].

The block-spin probability distribution $P_1(\boldsymbol{\mu})$ can be written in terms of a block-spin Hamiltonian

$$H_1(\boldsymbol{\mu}) = \sum_{\alpha} K_{\alpha,1} S_{\alpha}(\boldsymbol{\mu}). \qquad (5)$$

Here we assume that the spin operators S_{α} do not depend on the system size (i.e. they are all local and do not involve interactions between spins that are at a distance of the order of the system size), which is fine as long as the system size L is very large.

By performing n blocking procedure, one generates a flow of coupling constants $\mathbf{K} = \{K_{\alpha}\}$ given by the trajectory of $K_{\alpha,0} \to K_{\alpha,1} \to \ldots \to K_{\alpha,n}$, with $K_{\alpha,0} \equiv K_{\alpha}$ the microscopic couplings. This implements a Real Space Renormalization Group (RSRG) transformation on the couplings $\mathbf{K}_{n+1} = \mathcal{R}(\mathbf{K})$. Critical points of second-order phase transitions and scale-invariance are found by looking for the fixed point of the blocking procedure $K_{\alpha,\star}$, such that $\lim_{n\to\infty} K_{\alpha,n} = K_{\alpha,\star}$. Equivalently, we then have $P_n \to P_{\star}$, with

$$P_{\star}(\boldsymbol{\mu}) = \sum_{\boldsymbol{\sigma}} D(\boldsymbol{\mu}|\boldsymbol{\sigma}) P_{\star}(\boldsymbol{\sigma}). \qquad (6)$$

This is the hallmark of scale invariance: after sufficiently many coarse-graining steps, the probability distribution of the renormalized spins becomes stationary.

Obviously, the fixed point equation can only make sense if the system size is strictly infinite (such that the number of spins does not change after the blocking procedure). For very large but finite systems, P_n is expected to approach the fixed point probability distribution for $n \gg 1$ and $L/2^n \gg 1$, while finite size effects (and departure from the fixed point) will be generated for $L/2^n \simeq 1$.

The goal of the upscaling algorithm is to learn how to undo the blocking (i.e. downscaling) procedure. This is done by introducing another conditional probability $U(\boldsymbol{\sigma}|\boldsymbol{\mu})$ which attempts to undo the effect of $D(\boldsymbol{\mu}|\boldsymbol{\sigma})$.

Note that we do not require that it undoes it exactly, i.e. $\sum_{\boldsymbol{\mu}} U(\boldsymbol{\sigma}|\boldsymbol{\mu}) D(\boldsymbol{\mu}|\boldsymbol{\sigma}') = \delta_{\boldsymbol{\sigma},\boldsymbol{\sigma}'}$, nor do we aim for it, as it is impossible by construction since the information about the specific spin configuration in a block is lost. Instead, this should be (approximately) true at the distribution level, i.e. we aim for the best $U(\boldsymbol{\sigma}|\boldsymbol{\mu})$ such that

$$Q(\boldsymbol{\sigma}, \boldsymbol{\mu}) = U(\boldsymbol{\sigma}|\boldsymbol{\mu}) \sum_{\boldsymbol{\sigma}'} D(\boldsymbol{\mu}|\boldsymbol{\sigma}') P(\boldsymbol{\sigma}') \qquad (7)$$

is a good approximation of the true joint probability $P(\boldsymbol{\sigma}, \boldsymbol{\mu})$, i.e. $Q(\boldsymbol{\sigma}, \boldsymbol{\mu}) \simeq P(\boldsymbol{\sigma}, \boldsymbol{\mu})$. From Eq. (7), we find that given a block-spin configuration $\boldsymbol{\mu}$, with probability $P_1(\boldsymbol{\mu}) = \sum_{\boldsymbol{\sigma}} P(\boldsymbol{\sigma}, \boldsymbol{\mu}) = \sum_{\boldsymbol{\sigma}} D(\boldsymbol{\mu}|\boldsymbol{\sigma}) P(\boldsymbol{\sigma})$, we estimate the probability distribution of the underlying microscopic distribution to be $Q(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\mu}} Q(\boldsymbol{\sigma}, \boldsymbol{\mu}) = \sum_{\boldsymbol{\mu}} U(\boldsymbol{\sigma}|\boldsymbol{\mu}) P_1(\boldsymbol{\mu})$. That is, we are able to generate ("dream up") fine-grained configurations from coarse-grained ones.

Note that while the coarse-graining kernel D is independent of the (microscopic) couplings K_{α}, and thus of the temperature, it cannot be so for the upscaling kernel U. Indeed, for a given block spin, the typical underlying configurations of the (true) spins will be strongly temperature-dependent (quite random at high temperatures, very polarized at low temperatures), and this has to be learned during training. We focus here on criticality to test how well a system trained that way can generate scale-invariant configurations. This is indeed in principle possible since, if the kernel is general enough and properly trained, we should find that

$$P_{\star}(\boldsymbol{\sigma}) \simeq \sum_{\boldsymbol{\mu}} U(\boldsymbol{\sigma}|\boldsymbol{\mu}) P_{\star}(\boldsymbol{\mu}). \qquad (8)$$

As we shall see, already very simple kernels, as small as having three parameters, can achieve this with high fidelity. Furthermore, we will show how to generate faithful critical configurations without any input from Monte Carlo simulations (beyond training).

III. MODELS, TRAINING AND UPSCALING

A. Models

Machine learning and neural networks are obviously tailored for optimizing such problems and have been used recently in this context [21–23]. However, the complicated multilayer neural nets used there prevent a simple understanding of what has been learned. Here we focus instead on a very simple, one-layer convolutional network that already captures very well scale invariance and the renormalization group properties of block-spinning.

We choose a simple convolutional layer with a kernel that upscales a given $L/2 \times L/2$ configuration $\boldsymbol{\mu}$ as follows, see Fig. 2. We first upscale the configuration to a $L \times L$ configuration $\tilde{\boldsymbol{\sigma}}$ by transforming each up (down) spin to a block of four up (down) spins (this rule is known

as the "nearest-neighbor" rule in machine learning). This can be implemented using a kernel W_{nn}, with $\tilde{\boldsymbol{\sigma}} = W_{nn}\boldsymbol{\mu}$ (W_{nn} is a $L^2 \times (L/2)^2$ matrix). While this configuration $\tilde{\boldsymbol{\sigma}}$ could be a good guess at low temperatures (where spins in the same block are expected to point in the same direction), this is not so at higher temperatures (and in particular at criticality). To create correlations between spins belonging to different blocks, we use a convolutional network parameterized by a convolution kernel W_c (a $L^2 \times L^2$ matrix in general). Calling $W = W_c W_{nn}$ (matrix multiplication implied), the probability distribution of the spin σ_i given $\boldsymbol{\mu}$ is then given by

$$p(\sigma_i|\boldsymbol{\mu}) = \frac{e^{\sigma_i(W\boldsymbol{\mu})_i}}{\sum_{\sigma_i} e^{\sigma_i(W\boldsymbol{\mu})_i}}. \qquad (9)$$

Note that if σ_i was a binary variable (0 or 1), this probability would then be described by a sigmoid function). The conditional probability to upscale is thus given by $U(\boldsymbol{\sigma}|\boldsymbol{\mu}) = \prod_i p(\sigma_i|\boldsymbol{\mu})$.

We do not include a bias in Eq. (9) (i.e linear term $h_i\sigma_i$ in addition to the kernel, playing the role of a magnetic field), to preserve the $\mathbb{Z}_2$ symmetry of the problem. Furthermore, expecting spins to be more correlated with neighboring ones, we simplify the model even more by choosing W with a kernel of size k small, such that the configuration of a given spin is modified only depending on the configuration of the $k^2 - 1$ closest neighbors (see Fig. 2). We also impose the spatial symmetries of the system on the kernel, to respect invariance by $\frac{\pi}{2}$-rotations and reflections. Finally, as we focus on the Ising model with periodic boundary conditions, the system is translation invariant, and we use circular padding for the convolution kernel. In total, our model with kernel size k therefore consists of $(k+1)(k+3)/8$ independent parameters that will be optimized during the machine learning process. In the following, we will consider only kernels of sizes $k = 3$, 5, and 7, corresponding to 3, 6, and 10 free parameters. As we will see, the smallest kernel is already able to learn scale invariance and many properties of the renormalization group.

B. Training

The goal of the training procedure is to find the $U(\boldsymbol{\sigma}|\boldsymbol{\mu})$ that minimizes the Kullback-Leibler divergence D_{KL} between the true joint probability distribution $P(\boldsymbol{\sigma}, \boldsymbol{\mu})$ and the downscaled-upscaled one $Q(\boldsymbol{\sigma}, \boldsymbol{\mu})$, Eq. (7), i.e. minimizing

$$D_{KL}(P||Q) = \sum_{\boldsymbol{\sigma},\boldsymbol{\mu}} P(\boldsymbol{\sigma}, \boldsymbol{\mu}) \log\left(P(\boldsymbol{\sigma}, \boldsymbol{\mu})/Q(\boldsymbol{\sigma}, \boldsymbol{\mu})\right). \quad (10)$$

This is equivalent to minimizing the cross-entropy $H(P, Q) = -\sum_{\boldsymbol{\sigma},\boldsymbol{\mu}} P(\boldsymbol{\sigma}, \boldsymbol{\mu}) \log(Q(\boldsymbol{\sigma}, \boldsymbol{\mu}))$ and can be more conveniently rewritten as the minimization of the

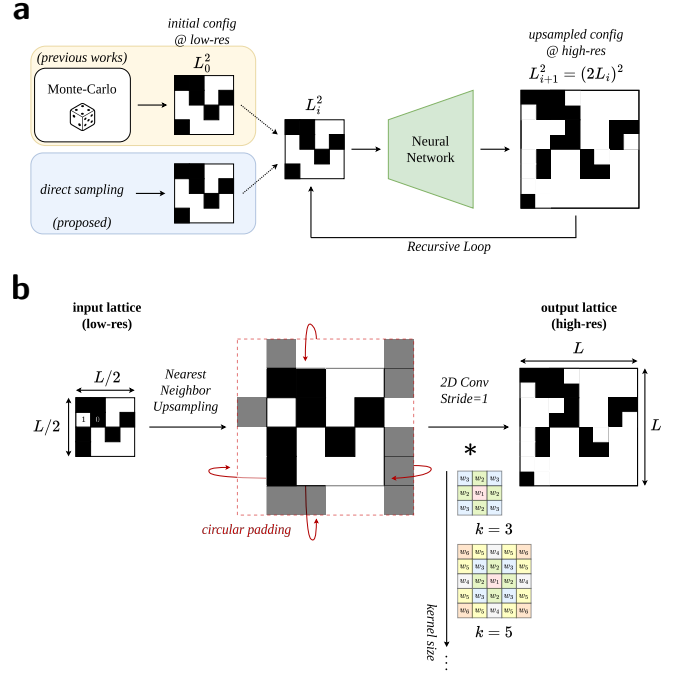FIG. 2. Iterative upscaling procedure (a) and proposed neural network architecture (b).

following loss function

$$\mathcal{L}(U) = -\sum_{\boldsymbol{\sigma}} P(\boldsymbol{\sigma}) \sum_{\boldsymbol{\mu}} D(\boldsymbol{\mu}|\boldsymbol{\sigma}) \log(U(\boldsymbol{\sigma}|\boldsymbol{\mu})), \qquad (11)$$

up to a constant independent of U. In practice, the average over all configurations is approximated by Monte Carlo (MC), using a cluster algorithm to ensure correct sampling at criticality [35], in terms of M configurations $\boldsymbol{\sigma}^{(a)}$, $a = 1, \ldots, M$. The training configurations $\boldsymbol{\sigma}^{(a)}$ are of size L_t^2. Assuming that $\boldsymbol{\sigma}^{(a)}$ corresponds to a given block-spin configuration $\boldsymbol{\mu}^{(a)}$ (of size $(L_t/2)^2$), we therefore have to optimize

$$\mathcal{L}(U) \simeq -\frac{1}{M} \sum_{a=1}^{M} \log\left(U(\boldsymbol{\sigma}^{(a)}|\boldsymbol{\mu}^{(a)})\right). \qquad (12)$$

If the block-spin procedure is not deterministic, as in the case of the majority rule used here, the sum over $\boldsymbol{\mu}$ in Eq. (11) is also sampled randomly, i.e. we choose one of the configuration $\boldsymbol{\mu}^{(a)}$ allowed from $\boldsymbol{\sigma}^{(a)}$ with probability $w_a = D(\boldsymbol{\mu}^{(a)}|\boldsymbol{\sigma}^{(a)})$. The loss function has to be weighted accordingly,

$$\mathcal{L}(U) \simeq -\frac{1}{M} \sum_{a=1}^{M} w_a \log\left(U(\boldsymbol{\sigma}^{(a)}|\boldsymbol{\mu}^{(a)})\right). \qquad (13)$$

Other than this change of loss, the training procedure is the same. In the following, we have trained using the true loss (with non-trivial weight w_a) and a loss where we impose a uniform weight $w_a = 1$ by hand.
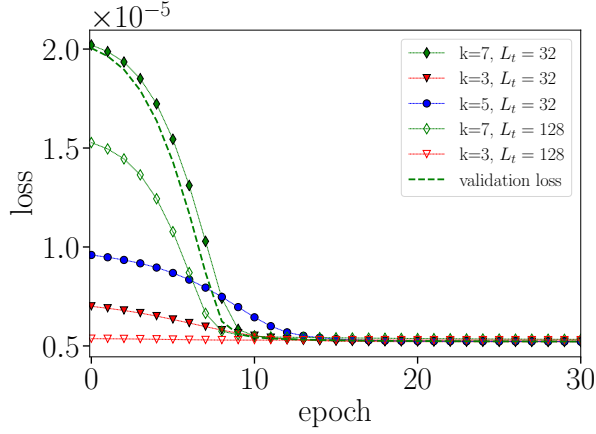
FIG. 3. Training losses for different models. The learning rate was in all cases 0.01, $M = 10^6$, and batch size 5×10^4. The dashed line shows a typical comparison between the training and the validation loss, here for $k = 7$, $L_t = 128$, and $M = 10^6$. As we train the models we test the loss on the $\frac{1}{5}M$ configurations that are statistically uncorrelated with the training configurations. The validation loss closely follows the training loss, showing that the model is not overfitting.

Hyper-parameter	Symbol
kernel size	k
linear size of training configurations	L_t
number of training configurations	M
weighting during training	w.
linear size of the initial configuration before upscaling	L_0
number of upscaling	n_{up}

TABLE I. List of hyper-parameters and their corresponding symbols.

The optimization procedure is performed using the AdamW algorithm with a fixed learning rate of 10^{-3}. The training minimizes the loss (13) of the generated spin configurations, averaged over M Monte Carlo samples. Gradient updates are computed over mini-batches, and no learning rate decay or early stopping is used.

C. Upscaling procedure

We have trained our simple neural nets on the two-dimensional Ising model, Eq. (1), at the critical point $K = \log(1 + \sqrt{2})/2$. The goal is to see whether such a simple neural net can capture and reproduce scale invariance, both qualitatively and quantitatively.

We can use the nets described above to generate "microscopic" configurations $\boldsymbol{\sigma}$ by upscaling a given "macroscopic" configuration $\boldsymbol{\mu}$. Indeed, the probability of $\boldsymbol{\sigma}$ given $\boldsymbol{\mu}$ is just given by the learned $U(\boldsymbol{\sigma}|\boldsymbol{\mu})$, and given
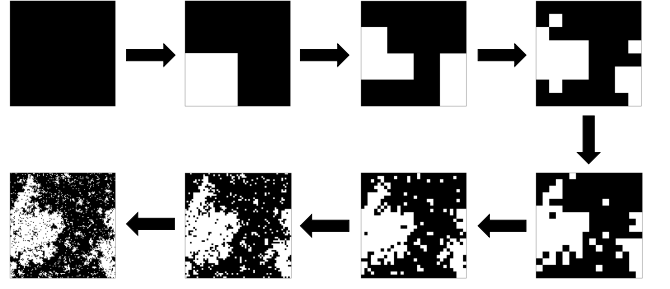

FIG. 4. Example of the upscaling procedure. The starting point is a single spin ($L_0 = 1$), top left. Each upscaling is symbolized by an arrow, doubling the system's size.

$\boldsymbol{\mu}$ we can sample a plausible configuration $\boldsymbol{\sigma}$ by simply sampling $U(\boldsymbol{\sigma}|\boldsymbol{\mu})$.

In previous works [21–23], the upscaling procedure was started from Monte-Carlo-sampled configurations $\boldsymbol{\mu}$ of finite size (for instance $L_0 = 16$). That is, the starting configuration $\boldsymbol{\mu}$ where first sampled using a Monte Carlo simulation (say at criticality), and then the (complicated) neural nets were used to create new macroscopic configurations $\boldsymbol{\sigma}$ of twice the size $L_1 = 2 \times L_0$. The procedure could then be repeated to obtain configurations of increasing sizes $L_n = 2^n L_0$.

Here instead, if not stated otherwise, the starting point is a single spin (i.e. $L_0 = 1$) randomly chosen to be ± 1 (with probability $1/2$). This single spin can then be upscaled n_{up} times to obtain configurations on lattices of size $L = 2^{n_{up}}$. We thus do not use the "help" of MC sampling to start from typical configurations at an intermediate size as previously done, but fully dream up configurations from one macroscopic block-spin, see Fig. 2 for the full procedure and Fig. 4 for an example of the generated configurations. For practical reasons, we have restricted upscaling up to $n_{up} = 7$ corresponding to lattices of size $L = 128$. For all models nets, and all $L = 2^{n_{up}}$, we generated 10^6 configurations.

The list of all the hyperparameters used for training and producing new configurations is summarized in Tab. I.

IV. FINITE SIZE SCALING OF GLOBAL OBSERVABLES

We start our analysis of upscaling with simple global observables that are typically used in numerical analysis of criticality (see e.g. Chapter 4 of [36]). Indeed, the finite-size scaling of these quantities allows for devising the scaling properties of the system at criticality.

From the upscaled configurations $\boldsymbol{\sigma}$ of size L, defining the magnetization density $m = L^{-2} \sum_i \sigma_i$ and energy density $e = L^{-2} \sum_{\langle i,j \rangle} \sigma_i \sigma_j$, we compute the magnetic susceptibility (χ), magnetization-energy cumulant (c_{me}),

heat capacity (c) and the Binder ratio (U_4)

$$\chi = L^2(\langle m^2 \rangle - \langle m \rangle \langle m \rangle),$$
$$c_{me} = L^2 \frac{(\langle |m||e| \rangle - \langle |m| \rangle \langle |e| \rangle)}{\langle |m| \rangle},$$
$$c = L^2(\langle e^2 \rangle - \langle e \rangle \langle e \rangle), \tag{14}$$
$$U_4 = 1 - \frac{\langle m^4 \rangle}{3\langle m^2 \rangle^2},$$

where $\langle \cdot \rangle$ is the average over the 10^6 upscaled configurations. At criticality, for the two-dimensional Ising model, those quantities have the well-known exact finite-size scaling behaviors

$$\chi \propto L^{\frac{\gamma}{\nu}},$$
$$c_{me} \propto L^{\frac{1}{\nu}}, \tag{15}$$
$$c \propto \ln(L),$$

where $\frac{\gamma}{\nu} = 2 - \eta = 7/4$, $\frac{1}{\nu} = 1$. Furthermore, U_4 tends to a universal constant the numerical estimate of which is $0.61069\ldots$ as $L \to \infty$ [37].

Figure 5 presents typical results for susceptibilities obtained with various model hyperparameter settings. The results are derived from upscaled data generated by single-layer models trained on $M = 10^6$ critical configurations of size $L_t = 32$. For comparison, we also include MC results obtained via the Swendsen-Wang cluster algorithm [35], obtained for the same system sizes (from 8 to 128) at criticality with the same statistics. This is sufficient statistics to ensure that the MC results are converged.

In general, there is good agreement between the upscaled and MC data across most quantities. Specifically, both the susceptibility, χ, and the magnetization-energy cumulant, c_{me}, exhibit clear power-law scaling, appearing as straight lines in the log-log plot. As could have been expected, increasing the convolution kernel size, k, results in systematic improvements, with model results converging more closely to the MC data. This effect is particularly pronounced for the heat capacity, which shows notable gains in accuracy with larger k values. However, accurately reproducing the logarithmic scaling behavior of the heat capacity proves more challenging. The upscaled data exhibits weak power-law behavior, which deviates slightly from the expected straight line in the log-normal plot, with the slope diminishing as k increases.

To quantitatively assess scaling behavior, we present in Table II the values of the critical exponents γ/ν and $1/\nu$, obtained by fitting data from lattices of size $L/4$ and L). These results from upscaled configurations are compared to those obtained from MC simulations. Increasing the kernel size k further enhances accuracy, with larger kernels yielding results closer to the exact exponents.

	γ/ν (32, 8)	γ/ν (64, 16)	γ/ν (128, 32)
MC	1.754	1.751	1.749
k=3	1.803	1.804	1.805
k=5	1.781	1.782	1.782
k=7	1.777	1.778	1.778
k=3, w.	1.834	1.835	1.836
k=5, w.	1.815	1.816	1.816
k=7, w.	1.812	1.812	1.812

	$1/\nu$, (32, 8)	$1/\nu$ (64, 16)	$1/\nu$ (128, 32)
MC	1.077	1.038	1.020
k=3	1.163	1.157	1.155
k=5	1.138	1.120	1.112
k=7	1.129	1.106	1.096
k=3, w.	1.169	1.164	1.163
k=5, w.	1.138	1.116	1.107
k=7, w.	1.126	1.103	1.093

TABLE II. Exponents γ/ν and $1/\nu$ determined from χ and c_{me} using the two values $(L, L/4)$ shown in parenthesis. The last digit displayed is reliable, given the statistics. The exact exponents are $\gamma/\nu = 1.75$ and $1/\nu = 1$.

While the results show promising convergence with the exact universal values, we observe some scatter across models with different hyperparameters, indicating potential for further refinement. To address this, we examined the influence of kernel size (k), training lattice size (L_t), and training set size (M) on the accuracy of estimated exponents. As shown in Fig. 6, which uses heat capacity as a representative metric, these factors have distinct effects.

For models with a kernel size of $k = 3$, increasing L_t or M has minimal impact on scaling accuracy, likely due to the limited number of free parameters constraining the network's learning potential. In contrast, models with a larger kernel size of $k = 7$ demonstrate greater adaptability. Increasing M significantly improves accuracy, suggesting the network benefits from a larger training set. However, increasing L_t beyond 32 does not yield systematic improvement; even at $L_t = 128$, variations in model architecture lead to more significant deviations than changes in lattice size. This suggests that beyond a certain lattice size, kernel size, and training set size are more influential in refining the accuracy of the critical exponents.

The results for the Binder ratio are comparable to that observed for the critical exponents of the susceptibilities, see Fig. 7. The scatter from the established reference value varies among the models, with a discrepancy of a few percent. To assess the reason of this discrepancy, we have also upscale data from 10^6 independent MC configurations simulations of a system of size $L_0 = 8$ in the spirit of [21–23], instead of a system of size $L_0 = 1$. This notably improves the Binder ratio to an error of less than a percent. However, upscaling from $L_0 = 8$ does not consistently lead to better determinations of the critical exponents from susceptibilities. Note that the value of
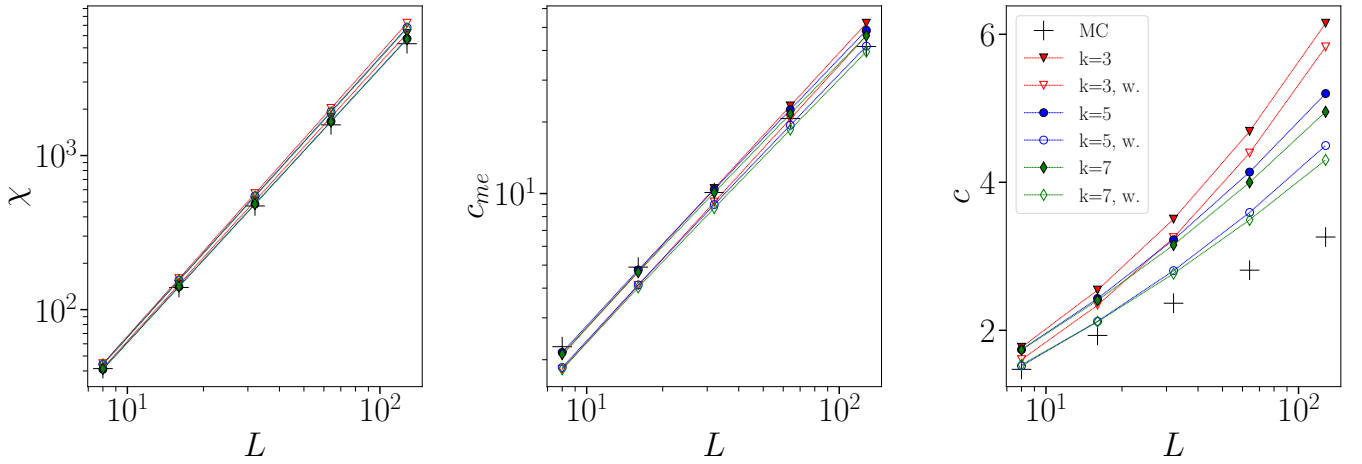
FIG. 5. Susceptibilities obtained from various hyper-parameters as a function of the upscaled size: a) magnetic susceptibility χ; b) magnetization-energy cumulant c_{me}; c) heat capacity c. Panels a) and b) are plotted in log-log scale (where power law behavior displays as linear) while panel c) is plotted in log-normal scale (where the $\ln L$ dependence displays as linear). All models were trained on $M = 10^6$ critical configurations of system size $L_t = 32$.
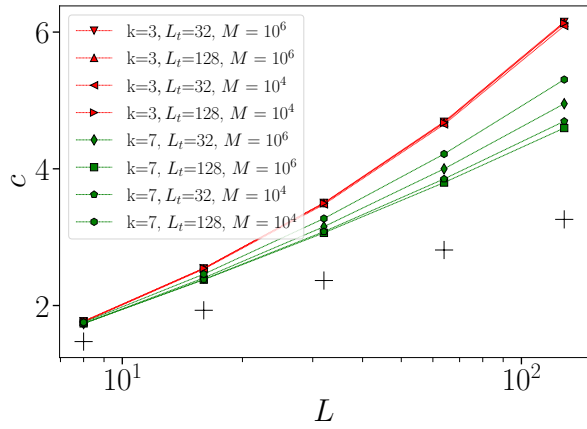

FIG. 6. Dependence of the heat capacity c calculated from the upscaled configurations on the size of the training configurations (L_t) and the statistics of the training set (M). Black crosses correspond to MC data.
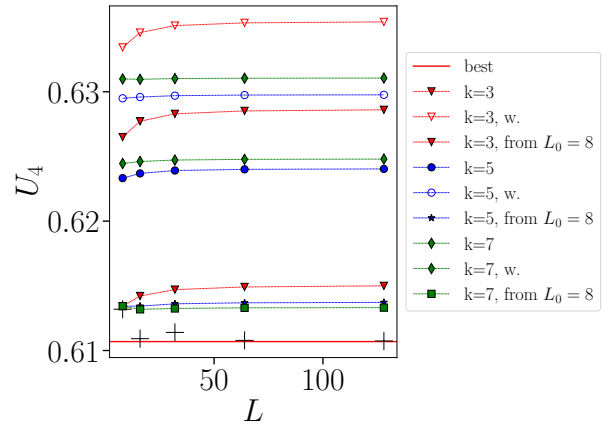
FIG. 7. Binder ratio computed from upscaled configurations and compared with the best-known value [37] and our Monte Carlo data (black crosses).

U_4 from the upscaled data from $L_0 = 8$ MC configurations is very close to the exact value of the Binder ratio at $L_0 = 8$, and almost independent of the number of upscaling. That is, it is as if the upscaling procedure does not change the value of the Binder ratio of the MC data, while still obeying scale invariance (since U_4 is a ratio of correlation functions that scale).

This can be verified by computing the probability distribution (PDF) of the order parameter $P_L(m)$. The true PDF obeys at criticality the scaling law (for sufficiently large L)

$$P_L(m) = L^{\beta/\nu}\tilde{P}(L^{\beta/\nu}m), \qquad (16)$$

with $\beta/\nu = 1/8$ for Ising 2D, and $\tilde{P}$ is a universal function that has been studied thoroughly both numerically and theoretically [38–44]. Note that the $P_L(m)$ obtained from MC simulations at $L = 128$ is converged up to better than 1% to the thermodynamic limit. Fig. 8 shows the logarithm of the PDF, $\log P_L(m)/P_L(0)$ obtained from upscaling system of size $L_0 = 1$. Remarkably, we see that it obeys the scaling law very well and is completely size-independent (i.e. independent of the number of upscaling). This explains why the correlation functions of the magnetization (such as χ) also obey their scaling laws very well. However, we also see that the shape of the PDF is slightly off (in particular in the middle), which explains why the Binder ratio is off. This is confirmed
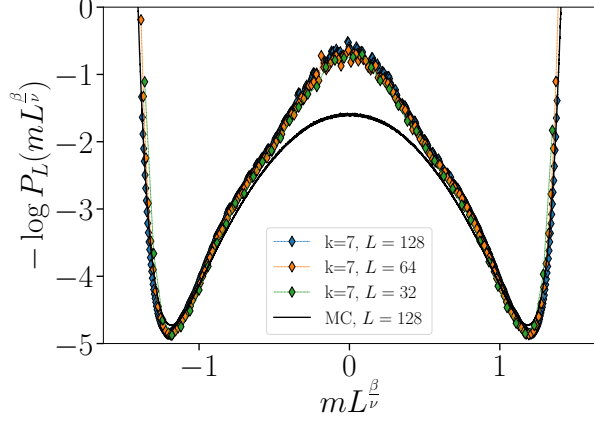
FIG. 8. Log of the probability distribution of the order parameter as a function of $mL^{\beta/\nu}$, obtained from MC simulations of lattice size $L = 128$, and upscaled configurations (upscaled to $L = 32$, $L = 64$ and $L = 128$) (hyperparameters: $k = 7$, $L_t = 32$, and $M = 10^6$). The probability is scale-invariant, as seen from the collapse of the three curve onto a single one. We use the exact exponent $\beta/\nu = 1/8$ for rescaling all the data.
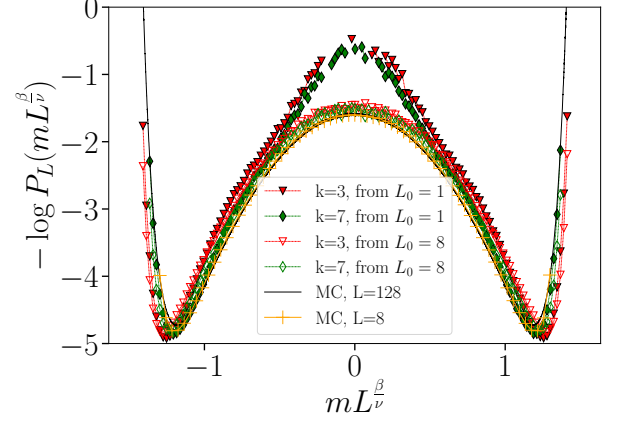
FIG. 9. Log of the probability distribution of the order parameter for a system of size $L = 128$, obtained from MC and upscale data. In the latter case, upscaling from $L_0 = 8$ instead of $L_0 = 1$ improves the results, whereas the kernel size is rather irrelevant. Models trained with $L_t = 32$ and $M = 10^6$. We use the exact exponent $\beta/\nu = 1/8$ for rescaling all the data.

by upsampling from MC configurations with $L_0 = 8$, see Fig. 9. We observe that the shape of the so-obtained PDF is much improved. This can be understood by remarking that the PDF has quite weak finite-size corrections, meaning that the PDF at $L_0 = 8$ from which the network upscale is already quite similar to that of a system of size $L = 128$. Therefore, the network, which as we have shown is good at generating scale invariant quantities, keeps the same global shape, with the proper scaling of the magnetization.

V. REAL SPACE RENORMALIZATION GROUP ANALYSIS OF UPSCALED CONFIGURATIONS

We have observed that susceptibilities obtained from the upscaled data yield highly stable results, effectively capturing global properties during the upscaling process. To subject the upscaled configurations to a more stringent evaluation, we conduct an RSRG analysis on them [45–53] . It is important to note that our objective is not to improve the RSRG block-spin transformation using machine learning, as previously done in [33]. Instead, we employ RSRG solely as a tool to scrutinize and analyze the properties of the upscaled configurations. In particular, we will use the same majority rule used throughout the manuscript to coarse-grain.

To analyze whether the neural network has learned the renormalization group properties associated with downscaling, we take inspiration from Monte Carlo RG (MCRG), which is a way to approximate the (exact)

RSRG transformation a la Kadanoff. Starting from the RSRG transformation defined in Sec. II, $\mathbf{K}_{n+1} = \mathcal{R}(\mathbf{K}_n)$ (where the index n denotes the number of blocking or the downscale step), with fixed point $\mathbf{K}_\star$ such that $\mathbf{K}_\star = \mathcal{R}(\mathbf{K}_\star)$, the scaling properties of the critical system can be obtained by analyzing the linearization of the RSRG transformation close to the fixed point,

$$K_{\alpha,n+1} - K_{\alpha,\star} = \sum_\beta \mathcal{T}^n_{\alpha,\beta}(K_{n,\beta} - K_{\beta,\star}). \qquad (17)$$

The matrix $\mathcal{T}$ gives us information about the flows near criticality and hence about the critical exponents. Diagonalizing it for $\mathbf{K}_n \simeq \mathbf{K}_\star$, the two largest eigenvalues e_1 and e_2 are related to the exponents $y_h = \frac{d+2-\eta}{2}$ and $y_\tau = \frac{1}{\nu}$ by $y_h = \frac{\ln e_1}{\ln 2}$ and $y_\tau = \frac{\ln e_2}{\ln 2}$, where 2 is the downscaling factor. Here the y_h exponent is related with an eigenvector which is odd and the exponent y_τ with an eigenvector which is even with respect to reflections of spin degrees of freedom.

The matrix $\mathcal{T}^n_{\alpha,\beta}$ can be determined from the correlation functions between different couplings since [47]

$$\mathcal{T}^n_{\alpha,\beta} = \frac{\partial K_{\alpha,n+1}}{\partial K_{\beta,n}} = \sum_\gamma \frac{\partial K_{\alpha,n+1}}{\partial \langle S_\gamma(\boldsymbol{\mu}) \rangle} \frac{\partial \langle S_\gamma(\boldsymbol{\mu}) \rangle}{\partial K_{\beta,n}}. \qquad (18)$$

Here $S_\gamma(\boldsymbol{\mu})$ is a spin operator in terms of a coarse-grained configuration $\boldsymbol{\mu}$ obtained from a finer-grained configuration $\boldsymbol{\sigma}$, and the average is done over the probability dis-

tribution of the latter. Using

$$\mathcal{M}^n_{\gamma,\beta} = \frac{\partial \langle S^n_\gamma(\boldsymbol{\mu}) \rangle}{\partial K_{\beta,n}} \tag{19}$$

$$= \langle S^n_\gamma(\boldsymbol{\mu}) S^n_\beta(\boldsymbol{\sigma}) \rangle - \langle S^n_\gamma(\boldsymbol{\mu}) \rangle \langle S^n_\beta(\boldsymbol{\sigma}) \rangle \tag{20}$$

$$\mathcal{N}^{n+1}_{\alpha,\gamma} = \frac{\partial \langle S^n_\gamma(\boldsymbol{\mu}) \rangle}{\partial K_{\alpha,n+1}} \tag{21}$$

$$= \langle S^{n+1}_\alpha(\boldsymbol{\mu}) S^n_\gamma(\boldsymbol{\mu}) \rangle - \langle S^{n+1}_\alpha(\boldsymbol{\mu}) \rangle \langle S^n_\gamma(\boldsymbol{\mu}) \rangle, \tag{22}$$

we then have $\mathcal{T} = \mathcal{M}^{-1}\mathcal{N}$. We immediately see that calculating this quantity entails calculating many different susceptibility-like quantities. Hence an analysis of the upscaled configurations using the matrix $\mathcal{T}$ constitutes a much more stringent test than simply calculating the susceptibilities of Eq. (14).

The advantage of this formulation is that the matrix $\mathcal{T}$ can be obtained from correlation functions (instead of an abstract transformation in coupling space). In practical calculations, three main factors limit the precision of the method: a) we are limited to calculation on finite lattices; b) one needs to choose the number of couplings to take into consideration; c) the number of configurations used to approximately compute the correlation functions is finite.

Despite these limitations, RSRG works extremely well on the 2D Ising model for the determination of the critical exponents. As a benchmark, we have used RSRG on lattices from $L = 32$ to $L = 1024$ for the 2D critical Ising model, with 10^6 MC configurations for each case. We have restricted the space of couplings to 36, i.e. all couplings with up to 4 spins that fit onto a 3×3 plaquette, see e.g. [49]. The results for the three most important universal exponents are given in Table III. If the $\mathcal{M}$ matrix was computed from couplings determined at n-th downscale step then the $\mathcal{N}$ matrix was computed from the couplings determined at n-th step and from the $n + 1$-th step. We find that even at $n = 1$, the couplings are already very close to the fixed point, i.e. $\mathbf{K}_1 \simeq \mathbf{K}_\star$. Furthermore, the exponents obtained from MCRG are very close to the exact ones, and in good agreement with the obtained from finite size scaling (Tab. II).

We now apply the same procedure to our models. For this, we upscale n_{up} times (starting from $L_0 = 1$) to compute the matrix $\mathcal{M}$, and then downscale once to compute the matrix $\mathcal{N}$, and then the matrix $\mathcal{T}$, using 10^6 configurations of upscaled configurations in total. The $\mathcal{T}$-matrix eigenvalues are shown in Figs. 10 a) to 10 d). The stability of the critical exponents obtained from finite-size scaling to upscaling is corroborated by this independent method. However, a slight scatter is observed, particularly compared to the RSRG from MC configurations obtained for similar lattice sizes. Notably, increasing the kernel size or the training system size does appear improve the results. This is summarized in Table IV, where we also recall the results obtained from finite-size scaling,

L	y_h (exact: 1.875)	$1/\nu$ (exact: 1)	ω (exact: -1)
32	1.87537(5)	0.931(2)	$-1.35(5)$
128	1.87582(4)	0.973(3)	$-1.08(5)$
512	1.87599(4)	0.992(2)	$-1.04(4)$
1024	1.87600(4)	0.997(2)	$-1.03(4)$

TABLE III. Exponents y_h, $y_\tau = \frac{1}{\nu}$, and ω determined from RSRG on lattices of various sizes using 10^6 MC configurations. The first two eigenvalues are easily identified as they are most relevant (as well as using their symmetry under spin flips). The leading correction ω can be identified by symmetry, however, it is higher in the spectrum, because several eigenvalues corresponding to redundant operators precede it. In the bracket, error bars on the last digit, given statistics of 10^6 MC configurations.

written in terms of the exponents η and ν. In particular, the exponents determined from RSRG are systematically different from those obtained by finite-size scaling (by a few percent for ν and $20-30\%$ for η). Whether increasing kernel sizes even more would resolve this tension remains an open question.

	η-FSS	η-RSRG	$1/\nu$-FSS	$1/\nu$-RSRG
MC	0.251	0.248	1.020	0.973
k=3	0.195	0.282	1.155	1.111
k=5	0.218	0.285	1.112	1.067
k=7	0.222	0.287	1.096	1.055
k=3, w.	0.164	0.243	1.163	1.130
k=5, w.	0.184	0.223	1.107	1.074
k=7, w.	0.188	0.227	1.093	1.059

TABLE IV. Critical exponents η and ν obtained from FSS and RSRG. FSS results from the last column of Table II. RSRG results are obtained from upscaled configurations to size $L = 128$.

Interestingly, our upscaling procedure also reconstructs the higher eigenvalues of the $\mathcal{T}$ matrix remarkably well. We give two examples that we see as relevant.

Fig. 10 on panel c) shows the exponent $y_{r,1} = \frac{ln e_3}{ln 2}$ derived from third eigenvalue of the matrix $\mathcal{T}$. We associate this eigenvalue with the most relevant redundant eigenvalue. Redundant eigenvectors and eigenvalues are non-universal and related to specifics of the renormalization procedure, such as e.g. the specific way in which we do downscale in the particular case. We have not been able to find the determination of this exponent in the literature. However, from our calculations, it seems to vanish as we increase the system size. Also note that only $k = 3$ results group near 0.1 while all larger kernels seem to group in the vicinity of zero.

Finally, Fig. 10 on panel d) shows the leading correction-to-scaling exponent ω, the exact value of which is -1. This exponent is determined from the 6th eigenvalue of the matrix $\mathcal{T}$, which in our calculation is related to the second most relevant eigenvector in the
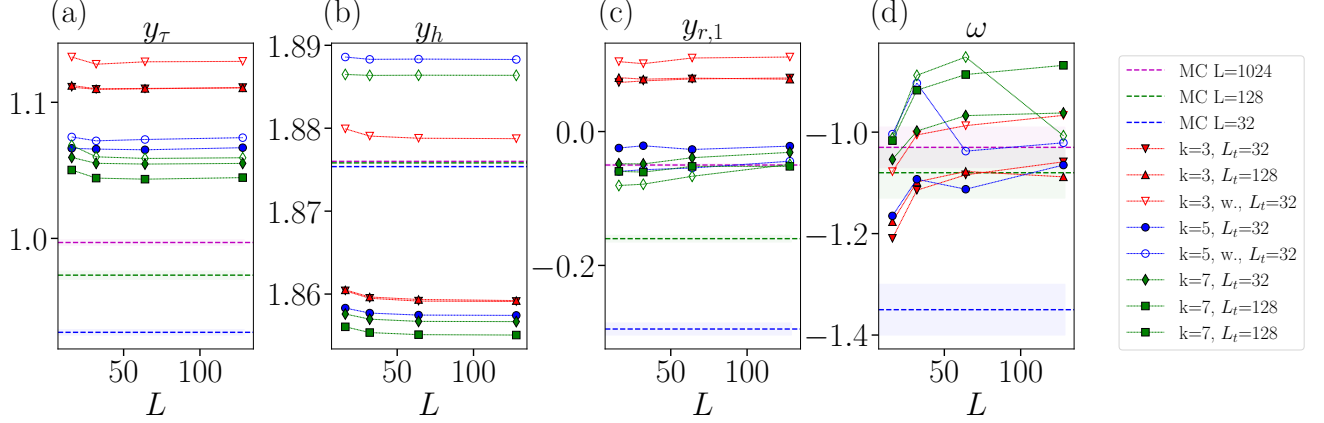
FIG. 10. Exponents obtained from the eigenvalues of the RSRG matrix $\mathcal{T}$ from the configurations upscaled from $L = 1$ to $L = 128$ and calculated on 10^6 configurations. Showing $y_\tau = 1/\nu$ on panel a); $y_h = \frac{d+2-\eta}{2}$, on panel b); most relevant redundant eigenvalue on panel c) and the correction-to-scaling exponent ω on panel d) . The shaded areas around the MC results give the statistical fluctuations, given the statistics of 10^6 MC configurations.

spectrum that is even under reflections of spin degrees of freedom. The determination of this eigenvalue from MC configurations becomes visibly affected by the statistics. This is also true for the upscaled data, as its value is not stable to upscaling. However, we see that the scatter is smaller than the difference between the exact value and by MC at small size ($L = 32$).

VI. IMPACT OF ARCHITECTURAL COMPLEXITY ON UPSCALING

Our results demonstrate that minimal, single-layer convolutional networks can successfully generate scale-invariant configurations that reproduce the universal features of critical phenomena. This naturally raises the question: *Does increasing model complexity improve performance?*

To address this, we considered more elaborate network architectures, specifically a two-layer, 32-channel local convolutional network similar in spirit to those used in prior work [21–23]. The architecture is as follows: i) nearest neighbor upscaling from $L/2$ to L as used in the single-layer models; ii) one convolutional layer of kernel size k from 1-to-32 channels of stride one and circular padding;[1] iii) a nonlinearity implemented with a ReLU; iv) one convolutional layer of kernel size k from 32-to-1 channels of stride one and circular padding. The upscaled configurations are then sampled using a sigmoid, similar to Eq. (9).

——————

[1] Here the convolutional kernels here are not symmetrized to implement the geometric symmetries of the lattice.

When trained to similar loss values, the two-layer network performs comparably to the single-layer models on global observables such as susceptibility and Binder ratio. However, we found that it consistently underperforms on quantities involving energy fluctuations, such as the heat capacity and magnetization-energy cumulant, see Fig. 11. This underperformance persists even when increasing the training set size or kernel width. While the root cause of this limitation is not yet fully understood, it suggests that the added depth and number of parameters do not necessarily improve the model's ability to reproduce fine thermodynamic structure.

Moreover, the expected benefits of overparameterization (such as improved generalization or better modeling of long-range correlations) are not observed. The network appears to behave similarly to small-kernel single-layer models in terms of both scaling exponents and probability distributions. This may indicate that the architecture's representational capacity is not effectively exploited during training, or that the essential features of criticality are already captured by local operations with minimal structure. Furthermore, the complexity of such architecture hinders the interpretability of the networks.

Our finding that increased network depth or complexity does not improve critical-point generation is consistent with observations in other contexts: simpler or symmetry-constrained models often capture the essential physics more effectively [54]. Highly parameterized networks often learn spurious, non-universal features, whereas minimalist "high-compressing" models or those explicitly encoding symmetries tend to focus on the relevant collective variables. Indeed, recent interpretability studies of phase-classifying neural networks reveal that such models internally learn representations of conventional order parameters [54, 55], reinforcing the idea
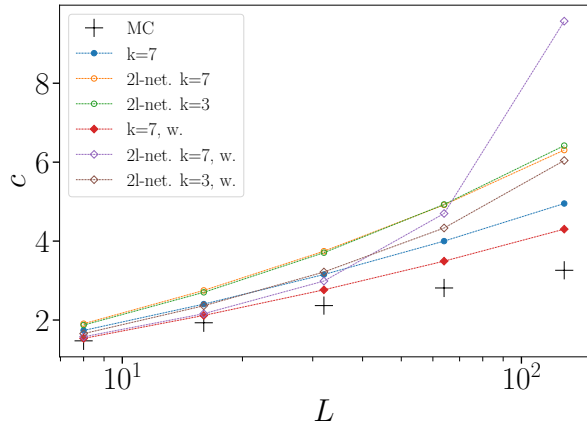
FIG. 11. Heat capacity from the upscaled configurations from the 2-layer model with kernel size $k = 3$ and $k = 7$, and with or without weight in the training. The results from the simpler model with the same kernel sizes are also shown for comparison. For all data $L_t = 32$ and $M = 10^6$.

from minimal input. Strikingly, networks with as few as three parameters are sufficient to reconstruct critical observables with quantitative accuracy and stable scaling exponents. Through both finite-size scaling and real-space renormalization group analysis, we demonstrated that the generated ensembles preserve key features of universality, including nontrivial RG eigenvalue spectra and very good probability distributions of the order parameter.

Perhaps most unexpectedly, increasing model complexity via additional convolutional layers and non-linearities does not improve performance and can even hinder the faithful reconstruction of critical statistics. This observation suggests that the essence of universality is encoded in simple, local, and symmetric transformation rules, much like those generating fractal structures. In this sense, our neural networks act as generative models for criticality, discovering scale-invariant ensembles from minimal assumptions and limited microscopic data.

These results open several directions. From a theoretical point of view, they raise the question of what it truly means to "invert" the RG flow: *Can such minimal models be understood analytically, or classified according to the universality class they encode?* From a practical perspective, they suggest new ways to simulate critical systems —without Hamiltonians or explicit dynamics— by learning the statistical structure of scale invariance directly. Extending this approach to higher dimensions, disordered systems, or systems out of equilibrium could shed new light on the generality of this framework and its analytical properties. In particular, introducing a handful of additional parameters into the model and training it to upscale at arbitrary temperatures could be an exciting avenue to understand how scale-invariance is encoded by the models.

that the crucial information governing a phase transition is remarkably low-dimensional. In our case, a three-parameter filter is sufficient to "discover" magnetization and critical fluctuations, much like Kadanoff's block-spin construction which distills large-scale order from local degrees of freedom.

While more advanced architectures (e.g. graph-based neural networks or attention mechanisms) could be applied to this problem, our results indicate that their added complexity is unnecessary for the clean 2D Ising critical point [28–30]. Incorporating known physical symmetries and locality constraints directly into a simple model appears to improve performance over a brute-force increase in model size.

This underscores a central message: complexity is not required to achieve scale-invariant generative modeling of critical systems. In fact, the essence of universality can be learned and generatively reproduced by surprisingly simple models, which offers advantages in interpretability, stability, and potential applicability to other systems.

VII. CONCLUSION

We have shown that simple, local neural networks can effectively learn to invert the renormalization group coarse-graining procedure in the two-dimensional Ising model, generating scale-invariant spin configurations

[1] K. G. Wilson and J. Kogut, Physics Reports **12**, 75 (1974).

[2] C. Bény, arXiv preprint (2013), arXiv:1301.3124 [quant-

ph].

[3] P. Mehta and D. J. Schwab, arXiv preprint (2014), arXiv:1410.3831 [stat.ML].

[4] E. De Mello Koch, R. De Mello Koch, and L. Cheng, IEEE Access **8**, 106487 (2020).

[5] A. Gordon, A. Banerjee, M. Koch-Janusz, and Z. Ringel, Physical Review Letters **126**, 240601 (2021).

[6] D. E. Gökmen, Z. Ringel, S. D. Huber, and M. Koch-Janusz, Physical Review Letters **127**, 240603 (2021).

[7] J. Erdmenger, K. T. Grosvenor, and R. Jefferson, SciPost Phys. **12**, 041 (2022).

[8] A. G. Kline and S. E. Palmer, New Journal of Physics **24**, 033007 (2022).

[9] G. Biroli, T. Bonnaire, V. de Bortoli, and M. Mézard, Nature Communications **15**, 9957 (2024).

[10] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová, Rev. Mod. Phys. **91**, 045002 (2019).

[11] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016) pp. 1874–1883.

[12] W. Yang, X. Zhang, Y. Tian, W. Wang, J.-H. Xue, and Q. Liao, IEEE Transactions on Multimedia **21**, 3106–3121 (2019).

[13] K. Fukami, K. Fukagata, and K. Taira, Journal of Fluid Mechanics **909**, A9 (2021).

[14] Y. Li, Y. Ni, R. A. C. Croft, T. D. Matteo, S. Bird, and Y. Feng, Proceedings of the National Academy of Sciences **118**, e2022038118 (2021).

[15] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, in *Advances in Neural Information Processing Systems*, Vol. 27, edited by Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger (Curran Associates, Inc., 2014).

[16] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," (2022), arXiv:1312.6114 [stat.ML].

[17] J. Ho, A. Jain, and P. Abbeel, in *Advances in Neural Information Processing Systems*, Vol. 33, edited by H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin (Curran Associates, Inc., 2020) pp. 6840–6851.

[18] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2017) p. 105–114.

[19] P. Esser, R. Rombach, and B. Ommer, in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2021) p. 12868–12878.

[20] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2022) p. 10674–10685.

[21] S. Efthymiou, M. J. Beach, and R. G. Melko, Physical Review B **99**, 075113 (2019).

[22] D. Bachtis, G. Aarts, F. Di Renzo, and B. Lucini, Physical Review Letters **128**, 081603 (2022).

[23] K. Shiina, H. Mori, Y. Tomita, H. K. Lee, and Y. Okabe, Scientific Reports **11**, 9617 (2021).

[24] D. Bachtis, Physical Review E **109**, 014125 (2024).

[25] D. Bachtis, Physical Review B **110**, L140202 (2024).

[26] L. Wang, Physical Review B **94**, 195105 (2016).

[27] S. J. Wetzel, Physical Review E **96**, 022140 (2017).

[28] Q. Ma, Z. Ma, J. Xu, H. Zhang, and M. Gao, Communications Physics **7**, 236 (2024).

[29] O. Kara, A. Sehanobish, and H. H. Corzo, arXiv preprint (2021), arXiv:2109.13925 [cs.CV].

[30] S. Y. Chang, M. Grossi, B. Le Saux, and S. Vallecorsa, in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)* (2023) pp. 229–235.

[31] M. Koch-Janusz and Z. Ringel, Nature Physics **14**, 578 (2018).

[32] P. M. Lenggenhager, D. E. Gökmen, Z. Ringel, S. D. Huber, and M. Koch-Janusz, Physical Review X **10**, 011037 (2020).

[33] J.-H. Chung and Y.-J. Kao, Physical Review Research **3**, 023230 (2021).

[34] Y. Wu and R. Car, Physical Review Letters **119**, 220602 (2017).

[35] R. H. Swendsen and J.-S. Wang, Physical Review Letters **58**, 86 (1987).

[36] D. P. Landau and K. Binder, *A Guide to Monte Carlo Simulations in Statistical Physics*, 4th ed. (Cambridge University Press, 2014).

[37] D. Nicolaides and A. D. Bruce, Journal of Physics A: Mathematical and General **21**, 233 (1988).

[38] K. Binder, Zeitschrift für Physik B **43**, 119 (1981).

[39] J.-P. Bouchaud and A. Georges, Physics Reports-review Section of Physics Letters **195**, 127 (1990).

[40] A. D. Bruce and N. B. Wilding, Physical Review Letters **68**, 193 (1992), number of pages: 0 Publisher: American Physical Society.

[41] X. S. Chen, V. Dohm, and N. Schultka, Physical Review Letters **77**, 3641 (1996), number of pages: 0 Publisher: American Physical Society.

[42] M. M. Tsypin and H. W. J. Blöte, Physical Review E: Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics **62**, 73 (2000), number of pages: 0 Publisher: American Physical Society.

[43] J. Xu, A. M. Ferrenberg, and D. P. Landau, Physical Review E: Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics **101**, 23315 (2020), number of pages: 6 Publisher: American Physical Society.

[44] I. Balog, A. Rançon, and B. Delamotte, Physical Review Letters **129**, 210602 (2022).

[45] D. Wallace and R. K. P. Zia, Physics Letters A **48**, 325 (1974).

[46] P. Peretto, Solid State Communications **19**, 235 (1976).

[47] R. H. Swendsen, Physical Review Letters **42**, 859 (1979).

[48] R. H. Swendsen, Physical Review Letters **52**, 1165 (1984).

[49] C. F. Baillie, R. Gupta, K. A. Hawick, and G. S. Pawley, Physical Review B **45**, 10438 (1992).

[50] A. Brandt and D. Ron, Journal of Statistical Physics **102**, 231 (2001).

[51] D. Ron, R. H. Swendsen, and A. Brandt, Physical Review Letters **89**, 275701 (2002).

[52] D. Ron, R. H. Swendsen, and A. Brandt, Physica A **346**, 387 (2005).

[53] D. Ron, A. Brandt, and R. H. Swendsen, Physical Review E **95**, 053305 (2017).

[54] C. Casert, T. Vieijra, J. Nys, and J. Ryckebusch, Phys. Rev. E **99**, 023304 (2019).

[55] R. C. Alamino, arXiv preprint (2024), arXiv:2402.11701 [cond-mat.dis-nn].