Tensor network to learn the wave function of data

Anatoly Dymarsky[1,2] and Kirill Pavlenko[2]

[1]*Department of Physics and Astronomy, University of Kentucky, Lexington, Kentucky 40506, USA*
[2]*Skolkovo Institute of Science and Technology, Skolkovo Innovation Center, Moscow 143026, Russia*

(Received 18 October 2021; accepted 14 July 2022; published 16 November 2022)

Tensor network architectures have emerged recently as a promising approach to various tasks of machine learning, both supervised and unsupervised. In this work we introduce a matrix product state-based network that simultaneously accomplishes the following two tasks: classification (discrimination) and sampling of visual data. We train the network using binary (black and white) version of MNIST, a data set of handwritten digits, to recognize as well as to sample images of a particular digit. We show our trained network is qualitatively representing the indicator function of the "full set" of all possible images of a given format depicting the particular digit. While the notion of the full set is difficult to define from the first principles, our construction provides a working definition, and we show that different ways to build and train the network lead to similar results. We emphasize, this means the trained network learns the "wave function of data," i.e., can be used to characterize the data itself, providing a novel tool to study global properties of the data sets of interest. First, using quantum mechanical interpretation we characterize the full set by calculating its entanglement entropy. Then we study its geometric properties such as mean Hamming distance, effective dimension, and size. The latter is the total number of images in binary black and white MNIST format which would be recognized as depicting a particular digit. Alternatively, it is the number of images of a given digit one would need to sample before the probability of sampling the same image twice would be of order one. While this number cannot be defined completely rigorously, we show its logarithm is largely independent of the way the network is defined and trained. We find that for different digits this number varies dramatically, from 2^{22} for digit 1 to 2^{92} for digit 8.

DOI: 10.1103/PhysRevResearch.4.043111

I. INTRODUCTION

Generalization is a remarkable ability of supervised learning algorithms to learn patterns underlying training data and subsequently perform well on new data sets. It reflects both potency of the algorithm but also certain simplicity of the training data itself. Namely, presence of patterns that might be apparent to a human eye but usually very difficult to quantify. On the contrary, data sets without underlying patterns, such as fully random or ad hoc ones can be learned but cannot be generalized [1,2]. To better understand when generalization is possible and inform development of more efficient supervised learning algorithms, it would be important to characterize patterns that underlie various data sets of interest. In this context a training set should be thought of as a small subset of the "full set of data," which includes all possible hypothetical data exhibiting given patterns. This picture serves as a motivation for our work, which proposes a practical method to learn and study the full set.

In this paper we introduce a novel tool, a tensor network that simultaneously can perform sampling and discrimination (classification) tasks. We argue this architecture learns the "wave function of data"—qualitatively, the indicator function of the full set, and thus provides new ways to quantitatively study and characterize it.

To keep the presentation simple, in what follows we focus on a particular example of MNIST, the data set of handwritten digits. All ideas and techniques can be immediately extended to other instances of supervised learning. MNIST contains images measuring 28 by 28 pixels; we transform them from grayscale to black and white for simplicity, such that there are 2^{784} possible images in total. A standard task would be to train a classifier to distinguish different digits. To further simplify things, we can train the network to recognize a particular digit, say digit 3, by distinguishing it from images of other digits, other symbols, or noise. Contemporary architectures can achieve this discrimination task with a small generalization error, which implies that among all possible 2^{784} images one can define the set of all images of digit 3, which our discriminator network would recognize. Good quality of generalization exhibited by various machine learning architectures [3,4] suggests this full set can be defined with a large degree of objectivity, essentially in the architecture-independent way.

Practically the full set is unfathomably large and is never available. In this work we propose a way to study it using

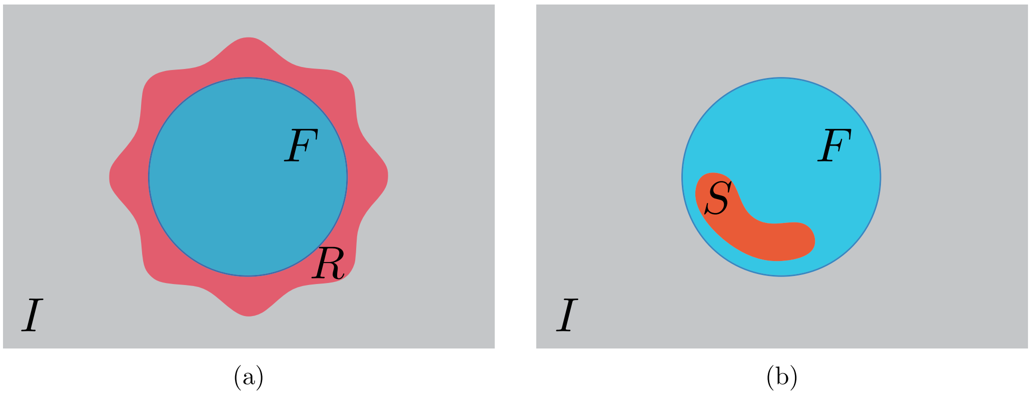


FIG. 1. On both panels I is the set of all 2^{784} images and F is the full set of images of digit i. Left panel: When Ψ is an ideal discriminator, it recognizes all images of digit i but may also recognize as i images of other digits or noise. This means $|\Psi(x)^2| \geqslant \epsilon$ for all $x \in F$, as well as for some set $x \in R$ depicted in red. In other words, support of (1) includes the full set F. Right panel: When Ψ is an ideal sampler, all sampled images are images of i. This means Ψ has the support on the subset of the full set $S \subset F$, while $|\Psi(x)|^2 \approx 0$ for $x \notin S$. When Ψ simultaneously accomplishes both discrimination and sampling tasks, left and right panels should be combined: the support of (1) coincides with the full set $R \approx S \approx F$.

a tensor network, which mathematically is an L_2-normalized "wave-function" $\Psi(x)$, defined on the space $I = \mathbb{Z}_2^{784}$ of all $N_I = 2^{784}$ possible images. Qualitatively, function

$$\mathcal{P}(x) := \begin{cases} 1, & \text{for } |\Psi(x)|^2 \geqslant \epsilon, \\ 0, & \text{for } |\Psi(x)|^2 < \epsilon, \end{cases} \tag{1}$$

with some appropriate ϵ, is the indicator function of the full set. To emphasize that Ψ characterizes the data itself and its properties exhibit robust independence of the tensor network architecture we call it the wave function of *data*. Using quantum mechanical interpretation of $\Psi(x)$ we can characterize the full set by calculating its entanglement entropy. We also study geometric properties of the full set such as mean Hamming distance, effective dimension, and the size. The latter is simply the approximate total number of images recognized by our network as depicting the given digit. In contrast to the first two properties, which can be studied using training set alone, size is the global property of the full set.

Before we proceed with the results, we will explain why tensor network sampler-discriminator/classifier is an appropriate architecture to define the full set via Eq. (1). We consider a hypothetical situation that network $\Psi(x)$ accomplishes both tasks almost ideally and deduce what that means for the support of Ψ. In the recent years tensor networks, such as matrix product states (MPS) and tensor trains, have been actively used to build various classification [5–8] and generative [9,10] algorithms. They demonstrate robust performance on par with the advanced CNN architectures [7,11]. In our case, we train Ψ for a particular digit i. Then the value $\mathcal{P}(x) = 1$ means Ψ recognizes x as an image of i.[1] Good quality of generalization means our network reliably recognizes images i outside of the training set. Now, assume Ψ is an ideal discriminator network. In this case, when Ψ has perfect quality of generalization, all images of i are recognized as such, but there still could be images of other digits or even noise recognized by our network as i. This means the support of Ψ includes but could be larger than the full set of i. This is illustrated in the left panel of Fig. 1. There gray square region represents all possible $N_I = 2^{784}$ images. Red area R represents images which our network "recognizes" as i, i.e., $|\Psi(x)|^2$ exceed ϵ for x from this area. The blue disk, denoted as F, represents the full set of images depicting given digit i. It is a subset of the red area R.

Next we proceed discussing $\Psi(x)$ as a sampler. It is convenient to think of $x = (x^1, \dots, x^{784})$ as an array of binary variables x^α. Tensor network architectures allow for an efficient evaluation of $|\Psi(x)|^2$ as a function of some components x^α while values of other components x^β being fixed. It therefore can be used for sampling: pixels are sampled consequently, using conditional probability distribution specified by $\Psi(x)$. This idea got traction recently and several such architectures were introduced in Refs. [9,10]. Clearly, only images with large values of $|\Psi(x)|^2$ can be sampled. To quantify the quality of sampling we train an auxiliary neural network of Ref. [12], which determines probability for the given sampled image of being the image of digit i. Provided our sampler $\Psi(x)$ achieves a good quality, i.e., ideally all sampled images depict i, we can think of $\Psi(x)$ as a function with the support on a subset of the full set. This is illustrated in the right panel of Fig. 1. There orange subset S of the blue disk represent images x for which $|\Psi(x)|^2$ is sufficiently large to be sampled, while for all other $x \notin S$, $|\Psi(x)|^2 \approx 0$.

The idea of the sampler-discriminator is to train an MPS-based tensor network $\Psi(x)$ which simultaneously accomplishes both tasks. As we saw above, the support of an ideal $\Psi(x)$ in this case would be equal to F, and (1) will be the indicator function of the full set. Schematically the training process looks as follows. We minimize the objective function

$$L = -\frac{1}{N_T} \sum_{x \in T} \ln |\Psi(x)|^2, \tag{2}$$

[1] For the classification task one needs to train ten networks $\Psi_i(x)$ for each i. Classification is then performed by maximizing $p_i = |\Psi_i(x)|^2 / \sum_j |\Psi_j(x)|^2$ over i; see Sec. IV.

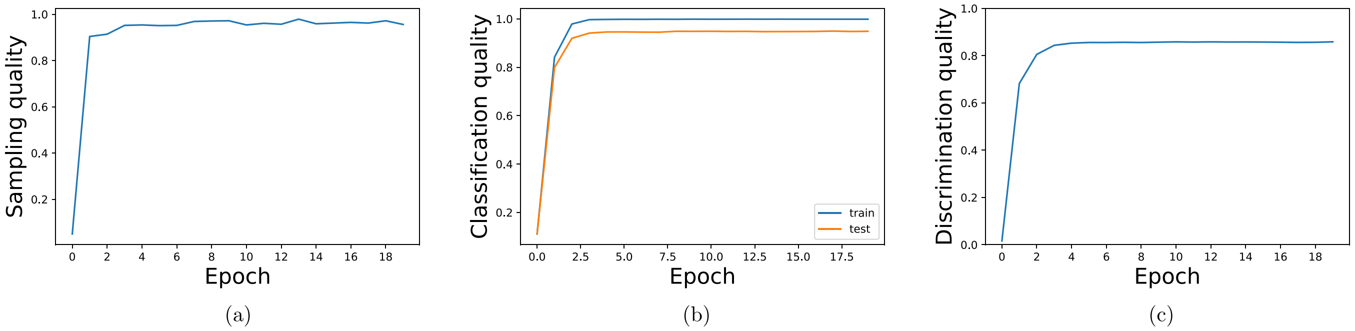


FIG. 2. (a) Quality of sampling by Ψ_3 with the bond dimension $D = 100$ during the training process. Quality is assessed by an auxiliary CNN [12]. (b, c) Quality of classification and discrimination during the training process by Ψ_i and Ψ_3 correspondingly, with the same $D = 100$.

where T represents the training set—a set of N_T images of digit i. It is a small subset inside the full set F. Importantly, our architecture enforces wave-function normalization

$$\sum_{x\in I} |\Psi(x)|^2 = 1, \quad I = \mathbb{Z}_2^{784}. \tag{3}$$

As a result decreasing of the loss function (2) automatically decreases value of $|\Psi(x)|^2$ for x outside of T. Assuming T is approximately uniformly distributed within F and $\Psi(x)$ changes smoothly, we may expect $|\Psi(x)|^2$ to mostly decrease outside of F, while inside F it would remain relatively large. The latter behavior would assure generalization of discriminator: value of $|\Psi(x)|^2$ for $x \in F$ would exceed certain threshold. The former property, smallness of $|\Psi(x)|^2$ for $x \notin F$, assures good quality of sampling. We therefore tend to conclude that a network trained this way will simultaneously accomplishes both discrimination and sampling with high quality, and thus will have a support on F, with Eq. (1) being its indicator function.

In practice decreasing of the loss function during training process will eventually lead to overfitting when $|\Psi(x)|^2$ is large for $x \in T$ but not necessarily for $x \in F$. We therefore stop training as soon as discrimination/classification begins to reduce after reaching its maximal value. The logic outlined above is schematic, we justify it *a posteriori* by examining the quality of recognizing (classifying) and sampling achieved by the trained Ψ. Further details of the network architecture and the training process are described in Sec. IV.

Ideally, for the trained network $\mathcal{P}$ defined in Eq. (1) is the indicator function of the full set: $|\Psi(x)|^2$ exceeds certain threshold for $x \in F$ and plunges below it for $x \notin F$. It therefore reflects the data itself rather than peculiarities of the architecture or the training process. To justify this claim we show that certain properties of $\Psi(x)$, such as quality of discriminating/classifying and sampling, typical value of $|\Psi(x)|^2$ for $x \in F$, value of entanglement entropy associated with $\Psi(x)$, etc., are not sensitive to MPS bond dimension or initialization seed. This confirms our main conclusion that the proposed architecture provides a novel way to quantitatively characterize the data itself, rather than peculiarities of the network design or the training process.

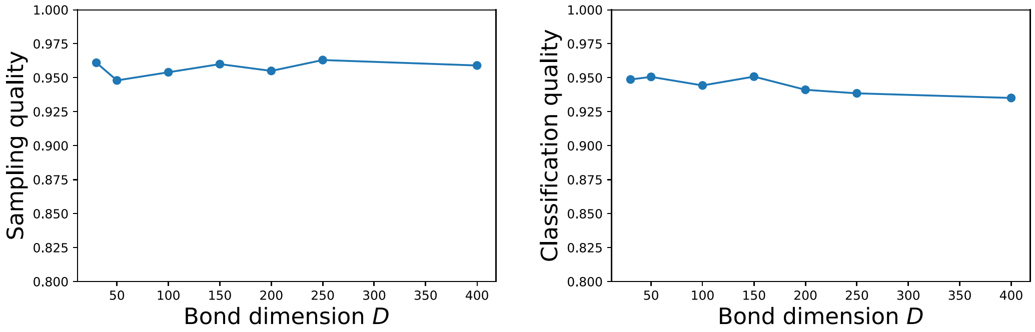
II. RESULTS

The core of our construction is the matrix product state real tensor network in the canonical form [13]. Mathematically it is a real-valued function $\Psi(x)$ where x^α is a vector of $28^2 = 784$ binary variables. Canonical form imposes normalization condition (3). We train the network by minimizing loss function (2) via gradient descent, and the test set T is the set of black and white MNIST images of digit i. Corresponding tensor network is labeled Ψ_i.

As the learning process proceeds, quality of sampling by Ψ_i gradually grows—the network remembers images from the training set and tries to replicate them. This is shown in the left panel of Fig. 2. The quality of recognizing digit i for images from the test set (calibration of threshold ϵ is discussed in Sec. IV) grows initially, but then may decay slightly due to overfitting. Similar behavior is exhibited by the quality of classification, for which all ten Ψ_i must be trained. This is shown in the right and central panels of Fig. 2. Overfitting becomes more pronounced when the bond dimension D of the tensor network grows. To prepare the network of interest, which would simultaneously accomplish both sampling and discrimination/classification tasks, the training process is stopped as soon as the quality of discrimination/classification reaches its maximum. For the sufficiently large $D \gtrsim 100$ this happens already after a few epochs.

We now demonstrate that core properties of properly trained Ψ_i are largely independent of the bond dimension D, provided the latter is sufficiently large, $D \gtrsim 30$. To begin with we study how the quality of sampling and classification depends on the bond dimension. The quality of classification is the maximal value from the central panel of Fig. 2, since we stop training at that point. Results for sampling and classification for different D shown in Fig. 3 confirm that quality remains essentially the same in a wide range of bond dimensions. It is also not sensitive to the initial seed.

Next we discuss to what extent Eq. (1) defines characteristic function of the full set of images of a given digit i. We also address the question of the size N_F of the full set—the global property of the full set which cannot be deduced directly from the training data set. In what follows we focus on $i = 3$ while results for other nine digits are qualitatively similar. First we would like to understand how many different images $x \in I$ there are with the given value of $|\Psi(x)|^2$ and what different values of $|\Psi(x)|^2$ represent. It would prove useful



(a)

(b)

FIG. 3. Quality of sampling (a) and classification (b) by the trained Ψ_3 as a function of bond dimension.

to use the language of statistical mechanics and think about different images $x \in I$ as possible microstates of some auxiliary physical system.[2] We define "energy" of a microstate $E(x) := -\ln|\Psi(x)|^2$ and entropy $S(E)$ such that $e^{S(E)}dE$ is the number of microstates with the energy between E and $E + dE$.[3] Then

$$\int_0^\infty e^{S(E)}dE = N_I. \tag{4}$$

In these notations normalization condition (3) becomes unity of the "partition function" at unit temperature

$$Z = \int_0^\infty e^{S(E)-E}dE = 1. \tag{5}$$

We first discuss the distribution of energies $E(x)$ of the training set, which we denote $\rho_{\rm tr}(E)dE$, shown in blue in Fig. 4. Minimization of loss function (2) is the minimization of energy averaged over $\rho_{\rm tr}$. The shape of $\rho_{\rm tr}$ is not robust and the mean value of the loss function $\langle E\rangle_{\rm tr}$ decreases with the increase of D. What remains essentially the same is the energy of the lowest states, which we define by averaging $|\Psi(x)|^2$ over the training set

$$E_0 = -\ln\langle e^{-E}\rangle_{\rm tr}. \tag{6}$$

The distribution $\rho_{\rm tr}$ should be compared with the distribution of energies for images sampled by the network itself. The probability of sampling an image x is equal $|\Psi(x)|^2$ and therefore the distribution of E for the sampled images is the Gibbs distribution at unit temperature

$$\rho_{\rm sm} = e^{S(E)-E}. \tag{7}$$

It is shown in red in Fig. 4. Naturally, $\langle E\rangle_{\rm sm}$ is larger than $\langle E\rangle_{\rm tr}$, i.e., the value of the loss function averaged over the set of sampled images is larger than the one for the training set. The shape of Eq. (7) is also changing with D. At the same time energy of the lowest states is robust and matches Eq. (6) with a good precision:

$$E_0 \approx -\ln\langle e^{-E}\rangle_{\rm sm}. \tag{8}$$

Moreover, with good accuracy it is equal to energy of the ground state $E_0 \approx E_g \equiv \min_x E(x)$, were minimization can go over training set $x \in T$ or the set of sampled images $x \in S$. From $e^S = \rho_{\rm sm}e^E$ we find that e^S is growing rapidly with E, at least for energies around $E \sim \langle E\rangle_{\rm sm}$. In other words there are exponentially more images x with larger values of $E(x)$. As E grows, quality of sampled images deteriorates. Qualitatively we can explain this as follows. Low-energy states x with $E(x) \approx E_0$ are the high quality "neat" images of digit 3, which will be recognized as such with almost 100% confidence. Each neat image gives rise to many more "corrupted" images, which still can be recognized as 3, albeit with a smaller confidence. These are the images with the larger values of E. As the level of corruption grows, so is the total number of such images, and their typical E increases. This is demonstrated in Fig. 5, where we show typical sampled images with three different values of E.

From this discussion it is clear there is no sharp value of threshold ϵ to define the boundary of the full set. The size of the full set, the total number of images x for which $\mathcal{P}(x) = 1$, is dominated by the images with $E \approx -\ln\epsilon$, which grows roughly as $e^E \approx 1/\epsilon$.

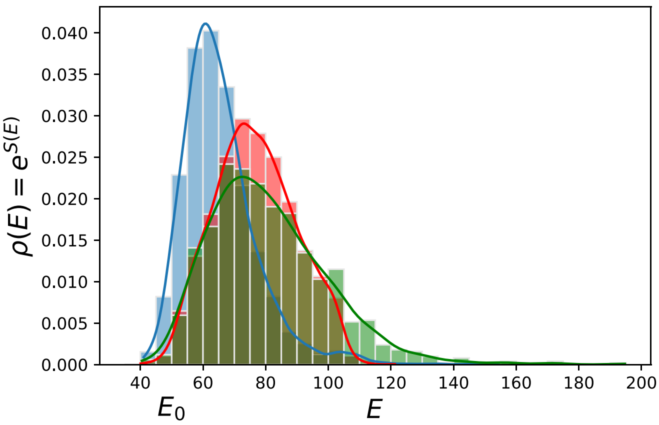


FIG. 4. Normalized distributions $\rho(E)dE$ of energies $E(x)$ for images x from the train (blue), sampled (red), and test (green) sets correspondingly. Solid lines are kernel density estimations.

[2] Familiarity with the basics of statistical mechanics are helpful but not necessary in what follows.

[3] As in statistical mechanics since microstates are discrete e^S is a sum of delta-functions, which can be approximated by a smooth function when the total number of states N_I is large, $N_I \gg 1$. In our case $N_I = 2^{784}$ and this condition is well satisfied.

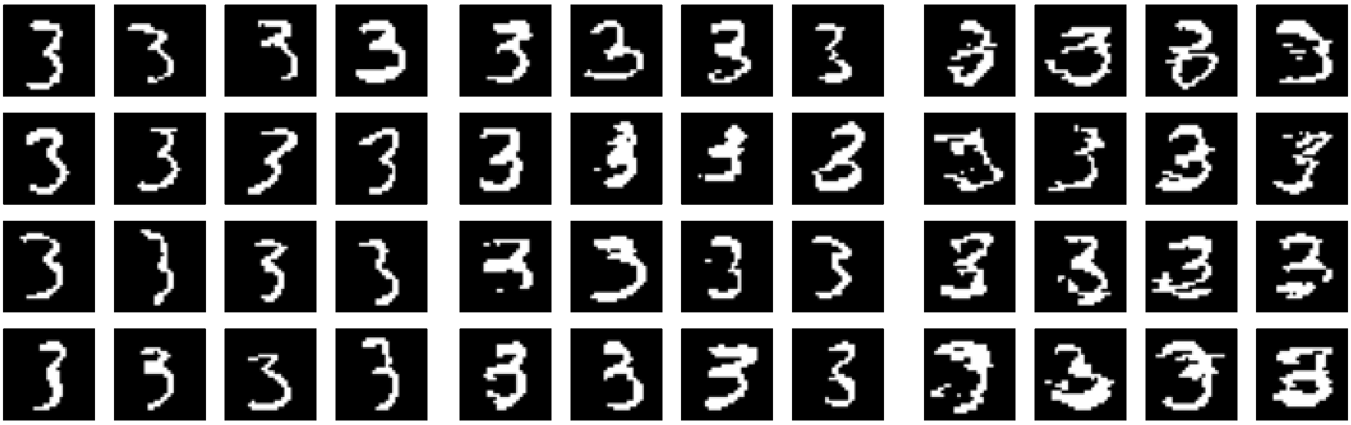


FIG. 5. Typical images sampled with help of Ψ_3 with three different ranges of E, from left to right $E = 60 \pm 5, 85 \pm 10, 125 \pm 10$.

To unambiguously define the size of the full set, we define the latter it to include only neat images of 3, in which case the threshold, which we will call ε, can be taken very close to $E_0 \approx E_{\rm g}$. We propose a way to fix ε in Sec. IV, but provided $\Delta E = \varepsilon - E_{\rm g}$ is small enough, at leading order the total size of the full set will be given by the exponent, while ΔE will control the subleading term,

$$N_F = \int_{E_{\rm g}}^{\varepsilon} e^S dE, \quad \ln N_F \approx E_0 + \ln(\rho_{\rm sm}(E_0)\Delta E) \approx E_0. \quad (9)$$

The number N_F can be interpreted as both, the leading exponent controlling the size of the full set—the total number of neat images of 3,

$$N_F \sim e^{E_0} \equiv 2^V, \quad V \equiv E_0/\ln 2, \quad (10)$$

as well as the number of images $M \approx N_F$ which need to be sampled before there would be repetitions, i.e., an image sampled twice. The latter interpretation follows from equating the total number of sampled images with given energy E, $M\rho_{\rm sm}(E)$ and the total number of images $e^{S(E)}$ with this energy. Understood as the equation on M, it yields $M = e^E$, where E should be larger than $E_{\rm g} \approx E_0$. Minimization of M over all possible value of E readily gives Eq. (10). An immediate question is how robust the latter interpretation is, given that $\rho_{\rm sm}$ depends on the network architecture. We test it by sampling images with one trained network, with the bond dimension D, and evaluating $E(x)$ using another trained network, with the bond dimension D'. Alternatively, we sample images using properly trained GAN. In all cases resulting $\rho_{\rm sm}$ have approximately the same value of $E_0 = -\ln\langle e^{-E}\rangle_{\rm sm}$ and therefore $N_F \sim M \sim e^{E_0}$ remain the same. The comparison of E_0 evaluated for Ψ_3 for different sets, training, test and sampled with help of Ψ_3 itself and with an auxiliary GAN, is shown in Fig. 6.

At this point, we conclude the energy of low-lying states E_0 is a robust characteristic of the full set which defines its size, thus answering the question from the abstract. Here the full set would be defined to include only neat images of 3 and similarly for other digits. The results for size V defined in Eq. (10) for all ten digits are shown in Table I.

Yet a closer look reveals $\mathcal{P}(x)$ of a trained network is not quite the characteristic function of the full set we hoped it would be. Looking at the distribution $\rho_{\rm test}$ of energies $E(x)$ for the images from the test set, we immediately find many neat images of 3 with the energies of order $\langle E\rangle_{\rm test}$, which are significantly larger than E_0. This is clear from Fig. 4, where $\rho_{\rm test}$ is shown in green. This indicates not a conceptual flaw but certain deficiency in how our network was trained. We argue now, for an idealized properly trained network typical values of $E(x)$ for images from both training and test sets should be around E_0. To confirm this we retrain our network by doubling the training set using GAN-generated images. As expected, as the size of the training set N_T increases both $\langle E\rangle_{\rm tr}$ and $\langle E\rangle_{\rm test}$ decrease, but the value of $E_0 = -\ln\langle e^{-E}\rangle$ defined with help of any set, train, test, or sampled, remains robust. The resulting picture is as follows. At leading order the total size of the full set is given by Eq. (10) and is accessible by a network trained with help of MNIST, while the number of neat images of 3 which our network misclassifies by assigning $\mathcal{P}(x) = 0$ is substantially smaller than e^{E_0}.

To further characterize the full set geometrically, we evaluate its mean distance and the effective dimension, defined with help of the Hamming distance. Individual images x are binary strings and Hamming distance $d(x_1, x_2)$, defined as the number of distinct components of x_1 and x_2, provides a simple

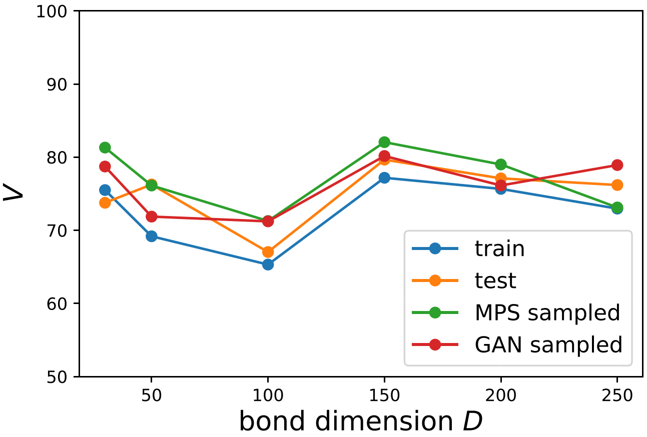


FIG. 6. The value of $V = E_0/\ln(2)$ evaluated for different sets: training and test sets as well as sets images sampled with Ψ itself and with help of an auxiliary GAN. All results are shown for trained $\Psi_3(x)$ as a function of bond dimension D.

TABLE I. The table summarizes main properties of the full sets associated with the digits of black and white MNIST. V, defined in Eq. (10), controls the size of the full set. $\langle d_{ab}\rangle$ is the mean Hamming distance. Δ is the full set effective (fractal) dimension. n is an average number of black pixels for images of a given digit i. $\bar{S}$ is the (averaged) entanglement entropy.

i	V	$\langle d_{ab}\rangle$	Δ	n	$\bar{S}$
0	90 ± 8	133 ± 3	13	138 ± 33	4.4 ± 0.2
1	22 ± 1	60 ± 25	4	60 ± 17	3.6 ± 0.1
2	84 ± 5	133 ± 27	12	118 ± 30	3.9 ± 0.2
3	72 ± 4	121 ± 29	12	112 ± 30	4.0 ± 0.1
4	69 ± 4	109 ± 26	11	96 ± 26	4. 1± 0.2
5	77 ± 4	127 ± 31	11	102 ± 30	3.8 ± 0.1
6	71 ± 4	115 ± 31	11	109 ± 30	3.8 ± 0.1
7	53 ± 2	99 ± 28	9	91 ± 24	3.7 ± 0.1
8	92 ± 4	124 ± 29	13	120 ± 31	4.0 ± 0.1
9	60 ± 5	105 ± 29	9	97 ± 26	3.8 ± 0.2

notion of distance between them. Clearly, Hamming distance is primitive in the sense it does not reflect how similar or different the essence of images would be to a human observer. Nevertheless, the full set understood as a subset of vertexes x of a unit cube equipped with the Hamming distance, satisfying $\mathcal{P}(x) = 1$ exhibits well-defined coarse grained geometric properties. The mean distance $d(x_1, x_2)$ between two random images of digit i, taken either from train/test or sampled sets is substantially smaller than two completely random images with the same mean value of black and white pixels. This is demonstrated in Table I where we show results for all ten digits.

To evaluate the full set effective dimension, we use the standard approach of Refs. [14,15]. For a set of images x_a we define minimal distance

$$d_{\min}(x_a) = \min_b d(x_a, x_b) \tag{11}$$

and then study how mean value $d = \langle d_{\min}(x_a)\rangle$, averaged over the set of x_a, scales with the set size K,

$$d \propto K^{-\Delta}. \tag{12}$$

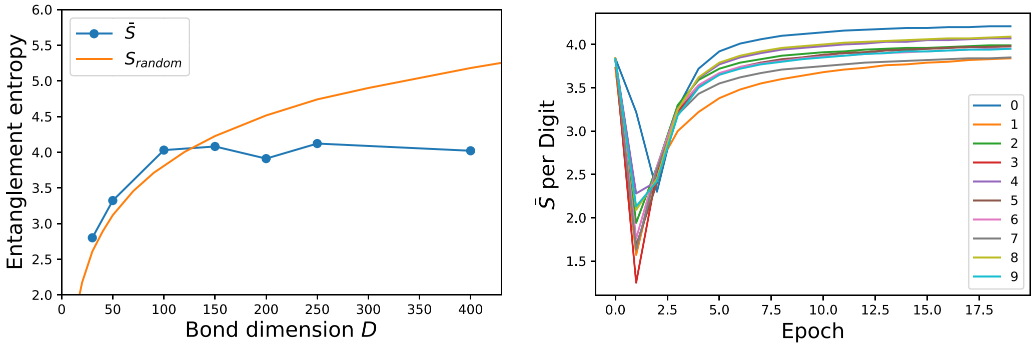
Here Δ is the effective (or fractal) dimension. Linear fit of $d(K)$ in log-log coordinates is shown in Fig. 8. We focus on neat images of digits i. Therefore, we consider a subset of train/test and sampled sets for which $E(x_a)$ is close to E_0. In both cases we obtain similar results, indicating full sets of digits have well-defined effective dimensions. The results are shown in Table I. We note, unlike the size of the full set, which is a global property requiring knowledge of the whole full set, mean distance and the effective dimensions can be inferred directly from the train/test set. Our results for Δ are compatible with previous studies of the effective dimension [14,15], which used Euclidean distance in conjunction with the original MNIST. This suggests rendering images black and white does not drastically affect geometries properties of the full set.

As an application of our architecture, we evaluate entanglement entropy (EE), by interpreting Ψ as a quantum-mechanical wave function. For a tensor network its maximal EE determines its expressiveness. In the context of quantum physics EE is a central quantity which measures the amount of information shared between different parts of the system [16]. In particular, it rigorously bounds classical mutual information associated with a bipartition [17]

$$I(A, B) \leqslant S_{AB}. \tag{13}$$

Popularity of EE transcended physics into machine learning with several different groups recently discussing it in the context of tensor-based architectures [18–22]. For a MPS architecture it is natural to discuss EE of bipartitions, S_k, where all $n = 784$ pixels are split into two groups of k and $n - k$ pixels correspondingly; see Sec. IV. Resulting *Page curve*[4] is shown in Fig. 9. We take EE averaged along the "plateau" region of bipartitions with k ranging between 200 and 600, for which S_k is approximately the same. This corresponds to splitting image horizontally into two halves. Averaged EE, which we denote by $\bar{S}$, slowly grows with epoch, which is expected: as the tensor network tries to better fit training data it needs more expressiveness associated with larger entanglement. Notably, after a few epochs $\bar{S}$ become essentially independent of the initial seed. We stop training Ψ when it exhibits maximal quality of discrimination/classification. Averaged $\bar{S}$ for such Ψ as a function of bond dimension is

[4] The name comes from the pioneering work evaluating EE of random states [23].



(a)

(b)

FIG. 7. (a) Average entanglement $\bar{S}$ of Ψ_3 as a function of bond dimension. (b) Average entanglement $\bar{S}$ for each Ψ_i during training.

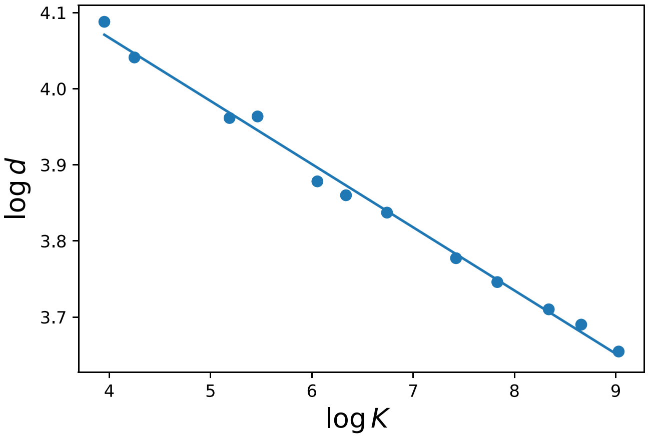


FIG. 8. Log-log plot for d as a function of number of samples K. Images are sampled with Ψ_3 with the bond dimension $D = 100$. Only images with $E(x) \approx E_0$ are included.

shown in the left panel of Fig. 7. It quickly grows for small D and becomes approximately constant for $D \gtrsim 100$. Robust independence of $\bar{S}$ on D and the initial seed is a further confirmation EE of a trained network is a characteristic of the data itself, not of the network architecture. This is the crucial difference between our work and previous studies of the EE in the context of tensor network algorithms. Our value of $\bar{S}$ should be contrasted with the maximal EE $S_{\max} = 2 \ln D$ for a tensor network with the bond dimension D, and for a random network with all MPS tensors drawn from the unitary ensemble, $S_{\text{random}} \simeq \ln D$; see Fig. 7.

Relatively small value of $\bar{S}$, and hence of mutual information between upper and lower halves of the images for all ten digits indicates there is a small number of ways to continue the image of a given digit, if a half of it is known. Schematically, this means there is a finite number of styles to handwrite any given digit i: once an upper part of the image is given, it fixes the digit itself and its style within a range of a few possibilities. This interpretation is corroborated by a positive correlation between the value of the entanglement, shown in Table I, and sizes of the full sets N_F; see Eq. (10). The logic here being that a larger number of styles will be reflected by a larger value of N_F.

The EE provides an upper bound on mutual information, an important information-theoretic properties characterizing the data. Recently mutual information and the entanglement entropy of data in the training set have been studied in Refs. [22,24–26]. Our work provides an alternative conceptually better way to evaluate it, as it is based on the full set, rather than the training set alone.

III. DISCUSSION

In this paper we introduced a tensor network architecture to learn the wave function of data Ψ. We introduced the notion of "the full set of data"—the collection of all hypothetical data exhibiting the same underlying pattern. As a main result, our tensor network provides a practical way to learn and subsequently characterize the full set by defining its indicator function $\mathcal{P}(x)$; see Eq. (1). We have trained Ψ using binary (black and white) version of MNIST and demonstrated its core properties are independent of the network parameters, which confirms they are characteristics of the data itself. Using Ψ we have estimated the sizes of the full sets of individual digits, i.e., the total number of black and white MNIST-style images depicting a particular digit i. The results are shown in Table I. To further illustrate utility of Ψ as a vehicle to study the data itself, we have calculated entanglement entropy, which upper bounds mutual information, associated with splitting images into two parts. The results are also shown in Table I. The entanglement entropy/mutual information of binary (black and white) MNIST images is small, which indicates relatively small number of different styles of handwriting.

The full set is a concrete, learnable cousin of an abstraction called the manifold of data in the series of recent papers [27–29]. The manifold of data, by definition, requires idealized "infinite data limit" when the training set grows indefinitely. This is in contrast with our approach suited for practically available data sets. We have seen in the case of MNIST digits, certain images from the train/test set fall outside of the full set defined via indicator function (1). We argued, this problem goes away as the size of the training set grows. In this case the distributions ρ_{tr}, ρ_{sm}, and ρ_{test} have smaller support and eventually, in the infinite data limit, collapse to a narrow vicinity of E_0. This is the limit in which the full set, which would strictly include all images of a given digit, would become the manifold of data of Refs. [27–29].[5] The rate with which the value of the loss function $\langle E \rangle_{\text{tr}} - E_{\text{g}}$ and $\langle E \rangle_{\text{test}} - E_{\text{g}}$ decreases with N_T—the size of the training set, as well its dependence on bond dimension D, should presumably follow the universal scaling laws outlined in Refs. [27–29]. To verify that would be an interesting problem.

One of the most important open questions of machine learning is to understand why certain data sets admit generalizations, as is the case of virtually any visual data set exhibiting a pattern recognizable by a human observer. It would be a substantial step forward to quantitatively characterize this phenomenon by explaining why generalization is possible. The studies of mutual information/entanglement entropy, initially motivated by a more narrow question of gauging suitability of tensor network-based architectures, is a first step in this direction. It is clear though, a much more comprehensive characteristic of data is necessary to understand if it admits generalization and the best way to achieve it. Our network is a novel tool to characterize the full set of data globally through the Boolean function $\mathcal{P}$, in the present case defined on a unit cube of the dimension $n = 784$. We surmise that the Boolean function complexity [30] as well as ideas developed in Ref. [31] could be the right language to characterize possible efficiency of generalization in the quantitative terms. Furthermore, drawing from the connection with quantum mechanics, we believe *circuit complexity* of Ψ interpreted as a quantum-mechanical wave function could be

[5] In our case the space of images I is discrete, while the data discussed in Ref. [28] is parametrized by vectors in $\mathbb{R}^n$. Hence, the full set in the infinite data limit will became a discrete version of the manifold of data.

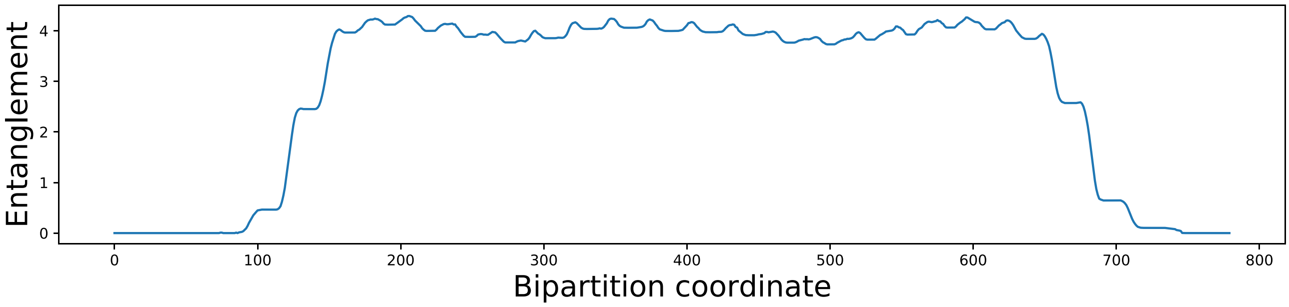


FIG. 9. Page curve for Ψ_3 with the bond dimension $D = 100$.

an important characteristic of patterns, underlying the initial data set.

IV. METHODS

A. MPS training procedure

We train a separate MPS $|\Psi_i\rangle$ (Fig. 10) for each label i in the data set ($i \in 0..9$ in case of MNIST). Our training procedure is similar to Refs. [9,10], with the key difference being the use of tangent-space gradient optimization of Ref. [32]. We start by mapping training samples x to product states $|x\rangle = \prod_{\otimes\alpha} |x^\alpha\rangle$ by representing each black or white pixel x^α with "down" $|0\rangle$ and "up" $|1\rangle$ states. With this representation of data samples we can define the probability of a given sample in accordance with Born's rules as

$$p(x) = \frac{|\langle x||\Psi\rangle|^2}{\langle\Psi||\Psi\rangle}. \qquad (14)$$

We train out network by minimizing values of the probability distribution $p(x)$ averaged over the training set T, while keeping normalization condition (3) for the wave function $\langle\Psi||\Psi\rangle = 1$. Namely, we minimize the negative-log likelihood (NLL) loss function L during training,

$$L = -\frac{1}{N_T}\sum_{x\in T}\log p(x). \qquad (15)$$

The gradient of the NLL loss function can be found analytically,

$$\frac{\partial L}{\partial|\Psi\rangle} = 2|\Psi\rangle - \frac{2}{N_T}\sum_{x\in T}\frac{|x\rangle}{\langle x||\Psi\rangle}. \qquad (16)$$

In practice we do not evaluate gradient with respect to $|\Psi\rangle$, instead we update each tensor of MPS independently via DMRG [9,10] method with a two-site update.

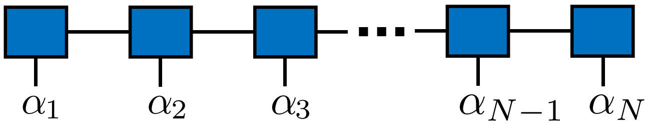


FIG. 10. Matrix product state graph representation. Blue boxes represent tensors (of rank 3) with α_i external indices, associated with pixel representation. Bond dimension D controls the maximum of other two dimensions of the tensors, depicted by horizontal lines.

Gradient descent is carried out by TSGO [32] with rotation angle (learning rate) $\eta = \pi/36$, which showed better and faster convergence compared to Adam or SGD.

1. Sampling

Sampling from a trained Ψ is carried via Born rule. Starting from the first pixel, which corresponds to the state $|x^1\rangle$, one samples it with marginal probability $p(x^1) = \frac{|\langle x^1||\Psi\rangle|^2}{\langle\Psi||\Psi\rangle}$. Here $\langle x^1||\Psi\rangle$ is a "state," i.e., a tensor with $N-1$ indexes $\alpha_2, \alpha_2, \dots, \alpha_N$. The subsequent pixel is sampled conditionally: $p(x^2|x^1) = p(x^1 \cup x^2)/p(x^1)$. To effectively calculate the probabilities one needs to keep the MPS in the right canonical form [9]. Typical examples of sampled images are shown in Fig. 5.

2. Quality of sampling

To assess quality of sampling we trained an auxiliary CNN to classify MNIST-like images on QMNIST [33], a data set which extends MNIST with additional 50K images, which allowed us to train CNN [12] on images never seen by our MPS. We sampled images with MPS and passed them through CNN which returned a probability of sampled image of, say, three to be classified as three. This probability we interpret as the quality of sampling, which is shown in the left panels of Fig. 2 and Fig. 3.

3. Classification

To classify images we take ten MPS Ψ_i trained to minimize the loss function (15) for each digit i separately. To predict label for an image x we calculate $\arg\max_i |\Psi_i(x)|^2$. Accuracy of such classification peaked at around 96% in our simulations, which is not as high as contemporary supervised NN architectures, but on par with common unsupervised methods.

4. Discrimination

Discrimination quality is the ability of a NN to distinguish images of, say, threes from any other images. In our case, due to normalization condition (3) the wave function Ψ_3 tends to decrease outside of the set of threes. In our setup $|\Psi(x)|^2 \geqslant \epsilon$ indicates that x is contained in the set of threes and x is outside of the set of threes if $|\Psi(x)|^2 < \epsilon$. Here we propose a simple method to fix ϵ and estimate discrimination quality. While assessment of discrimination quality in principle requires to test MPS against the whole "space of images" I, in practice, most of this space is irrelevant. Our model does not confuse threes with random images. The hardest examples to discriminate are pictures which are structurally similar to threes, i.e., those depicting other symbols. That is why we test Ψ_i on other digits on MNIST. We split training test into two parts. One part is used to train the model, while the other is used to fix ϵ such that the balanced accuracy on the test set is maximized. The resulting quality of discrimination is shown on the right panel of Fig. 2.

5. Fixing ε

The value of ε, introduced around Eq. (9) should be fixed to separate "neat" images of digit i from the "corrupted" ones. There is no precise definition of "neat" or "corrupted." However, we can try to estimate ε as follows. First, we use trained Ψ_i to sample the images of i and keep only those with the value of $E \approx E_0$. With the help of auxiliary CNN, we estimate sampling quality of each sample and calculate mean and variance of the sampling quality distribution. Then, we start looking into sampled images with larger values of $E > E_0$. For the given E we estimate mean and variance of the sampling quality distribution. Thus for each E we have mean value and variance. Now we are looking for $E = E_*$ which is large enough such that corresponding sampling quality distribution is sufficiently different from the one for $E = E_0$, namely, corresponding mean values are separated by at least the standard deviation. This value of E_* is taken to be $\varepsilon = E_*$. We have checked numerically that ε obtained in this way yields sub-leading contribution to N_F; see Eq. (10).

6. Further comments

We trained an auxiliary deep convolutional GAN to independently sample additional MNIST-like images. With these samples we doubled the amount of images and retrained MPS on extended data. As a result the value of mean $\langle E \rangle$ averaged over the test set decreased by 4%. Additionally, we used GAN samples to verify our estimation of effective (fractal) dimension for each digit. The results are in agreement with values of Δ from the Table I.

B. Entanglement entropy

Entanglement entropy of a bipartition which splits a quantum spin-chain into two parts A and B consisting of k "left" and $n-k$ "right" spins correspondingly is defined as

$$S(\rho_A) = -\mathrm{Tr}(\rho_A \log \rho_A) = -\mathrm{Tr}(\rho_B \log \rho_B) = S(\rho_B), \quad (17)$$

where $\rho_A = \mathrm{Tr}_B(\rho_{AB})$ and $\rho_B = \mathrm{Tr}_A(\rho_{AB})$ are reduced density matrices for each partition and $\rho_{AB} = |\Psi\rangle\langle\Psi|$ for a pure state.

For any pure state the entanglement entropy can be expressed using the singular values of the Schmidt decomposition of the state,

$$|\Psi\rangle = \sum_i \lambda_i |u_i\rangle_A \otimes |v_i\rangle_B, \quad (18)$$

where $|u_i\rangle_A$ and $|v_i\rangle_B$ are orthonormal states of the subsystems A and B separated. The entanglement entropy reduces to

$$S(\rho_A) = S(\rho_B) = -\sum_i |\lambda_i|^2 \log(|\lambda_i|^2). \quad (19)$$

The advantage of the MPS is that we can bring network to the form with the orthogonality center to be the bond between the subsystems A and B. Then Schmidt decomposition (18) becomes singular value decomposition of the product of two MPS tensors from the nodes adjacent to the orthogonality center bond [13].

ACKNOWLEDGMENTS

We thank Vassilis Anagiannis, Miranda Chen, and especially Vasily Pestun for collaboration at the early stages of this project. We also acknowledge helpful discussions with Dan Roberts, Ivan Oseledets, Miles Stoudenmire, and David Schwab. A.D. acknowledges support from the National Science Foundation under Grant No. PHY-2013812.

[1] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, Understanding deep learning requires rethinking generalization, arXiv:1611.03530 (2017).

[2] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, Understanding deep learning (still) requires rethinking generalization, Commun. ACM **64**, 107 (2021).

[3] B. Neyshabur, S. Bhojanapalli, D. McAllester, and N. Srebro, Exploring generalization in deep learning, in *Advances in Neural Information Processing Systems* (NIPS, Long Beach, CA, 2017), Vol. 30, https://proceedings.neurips.cc/paper/2017/file/10ce03a1ed01077e3e289f3e53c72813-Paper.pdf.

[4] K. Kawaguchi, L. P. Kaelbling, and Y. Bengio, Generalization in deep learning, arXiv:1710.05468 (2020).

[5] E. Stoudenmire and D. J. Schwab, Supervised learning with tensor networks, in *Advances in Neural Information Processing Systems*, Vol. 29, edited by D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (Curran Associates, Red Hook, NY, 2016).

[6] A. Novikov, M. Trofimov, and I. Oseledets, Exponential machines, Bulletin of the Polish Academy of Sciences: Technical Sciences **66**, 789 (2017).

[7] S. Efthymiou, J. Hidary, and S. Leichenauer, Tensornetwork for machine learning, arXiv:1906.06329 (2019).

[8] V. Anagiannis and M. C. N. Cheng, Entangled q-convolutional neural nets, Mach. Learn.: Sci. Technol., **2**, 045026 (2021).

[9] Z.-Y. Han, J. Wang, H. Fan, L. Wang, and P. Zhang, Unsupervised Generative Modeling Using Matrix Product States, Phys. Rev. X **8**, 031012 (2018).

[10] S. Cheng, L. Wang, T. Xiang, and P. Zhang, Tree tensor networks for generative modeling, Phys. Rev. B **99**, 155131 (2019).

[11] I. Glasser, R. Sweke, N. Pancotti, J. Eisert, and J. I. Cirac, Expressive power of tensor-network factorizations for probabilistic modeling, with applications from hidden Markov models to quantum machine learning, *Proceedings of the 33rd*

International Conference on Neural Information Processing Systems (2019).

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, Imagenet classification with deep convolutional neural networks, *Proceedings of the 25th International Conference on Neural Information Processing Systems* (ACM, New York, NY, 2012).

[13] G. Vidal, Efficient Classical Simulation of Slightly Entangled Quantum Computations, Phys. Rev. Lett. **91**, 147902 (2003).

[14] M. Hein and J.-Y. Audibert, Intrinsic dimensionality estimation of submanifolds in Rd, *Proceedings of the 22nd International Conference on Machine Learning* (ACM, New York, NY, 2005).

[15] E. Facco, M. d'Errico, A. Rodriguez *et al.*, Estimating the intrinsic dimension of datasets by a minimal neighborhood information, Sci. Rep. **7**, 12140 (2017).

[16] J. Eisert, M. Cramer, and M. B. Plenio, Colloquium: Area laws for the entanglement entropy, Rev. Mod. Phys. **82**, 277 (2010).

[17] J. Um, H. Park, and H. Hinrichsen, Entanglement versus mutual information in quantum spin chains, J. Stat. Mech.: Theory Exp. (2012) P10026.

[18] V. Pestun and Y. Vlassopoulos, Tensor network language model, arXiv:1710.10248 (2017).

[19] V. Pestun, J. Terilla, and Y. Vlassopoulos, Language as a matrix product state, arXiv:1711.01416 (2017).

[20] Y. Levine, D. Yakira, N. Cohen, and A. Shashua, Deep learning and quantum entanglement: Fundamental connections with implications to network design arXiv:1704.01552 (2017).

[21] Y. Levine, Or Sharir, N. Cohen, and A. Shashua, Quantum Entanglement in Deep Learning Architectures, Phys. Rev. Lett. **122**, 065301 (2019).

[22] J. Martyn, G. Vidal, C. Roberts, and S. Leichenauer, Entanglement and tensor networks for supervised image classification, arXiv:2007.06082 (2020).

[23] D. N. Page, Average Entropy of a Subsystem, Phys. Rev. Lett. **71**, 1291 (1993).

[24] D.-L. Deng, X. Li, and S. Das Sarma, Quantum Entanglement in Neural Network States, Phys. Rev. X **7**, 021021 (2017).

[25] I. Convy, W. Huggins, H. Liao, and K. Birgitta Whaley, Mutual information scaling for tensor network machine learning, Machine Learning: Science and Technology **3**, 015017 (2021).

[26] S. Lu, M. Kanász-Nagy, I. Kukuljan, and J. Ignacio Cirac, Tensor networks and efficient descriptions of classical data, arXiv:2103.06872 (2021).

[27] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, Scaling laws for neural language models, arXiv:2001.08361 (2020).

[28] U. Sharma and J. Kaplan, A neural scaling law from the dimension of the data manifold, arXiv:2004.10802 (2020).

[29] Y. Bahri, E. Dyer, J. Kaplan, J. Lee, and U. Sharma, Explaining neural scaling laws, arXiv:2102.06701 (2021).

[30] S. Jukna, *Boolean Function Complexity* (Springer, Berlin, 2012).

[31] Y. Huang, Provably efficient neural network representation for image classification, arXiv preprint arXiv:1711.04606 (2017).

[32] Z.-Z. Sun, S.-J. Ran, and G. Su, Tangent-space gradient optimization of tensor network for machine learning, Phys. Rev. E **102**, 012152 (2020).

[33] C. Yadav and L. Bottou, Cold case: The lost MNIST digits, In *Advances in Neural Information Processing Systems 32* (Curran Associates, Red Hook, NY, 2019).