Phases of learning dynamics in artificial neural networks in the absence or presence of mislabeled data

MACHINE LEARNING
Science and Technology

OPEN ACCESS

RECEIVED
28 December 2020

REVISED
18 March 2021

ACCEPTED FOR PUBLICATION
7 April 2021

PUBLISHED
19 July 2021

PERSPECTIVE

Phases of learning dynamics in artificial neural networks in the absence or presence of mislabeled data

Yu Feng[1,2] and **Yuhai Tu**[1,*]

[1] IBM T. J. Watson Research Center, Yorktown Heights, NY 10598, United States of America
[2] Department of Physics, Duke University, Durham, NC 27710, United States of America
[*] Author to whom any correspondence should be addressed.

E-mail: yuhai@us.ibm.com

Keywords: neural network, stochastic gradient descent, learning dynamics, statistical physics, phase transition, order parameter

Supplementary material for this article is available online

Abstract

Despite the tremendous success of deep neural networks in machine learning, the underlying reason for their superior learning capability remains unclear. Here, we present a framework based on statistical physics to study the dynamics of stochastic gradient descent (SGD), which drives learning in neural networks. Using the minibatch gradient ensemble, we construct order parameters to characterize the dynamics of weight updates in SGD. In the case without mislabeled data, we find that the SGD learning dynamics transitions from a fast learning phase to a slow exploration phase, which is associated with large changes in the order parameters that characterize the alignment of SGD gradients and their mean amplitude. In a more complex case, with randomly mislabeled samples, the SGD learning dynamics falls into four distinct phases. First, the system finds solutions for the correctly labeled samples in phase I; it then wanders around these solutions in phase II until it finds a direction that enables it to learn the mislabeled samples during phase III, after which, it finds solutions that satisfy all training samples during phase IV. Correspondingly, the test error decreases during phase I and remains low during phase II; however, it increases during phase III and reaches a high plateau during phase IV. The transitions between different phases can be understood by examining changes in the order parameters that characterize the alignment of the mean gradients for the two datasets (correctly and incorrectly labeled samples) and their (relative) strengths during learning. We find that individual sample losses for the two datasets are separated the most during phase II, leading to a data cleansing process that eliminates mislabeled samples and improves generalization. Overall, we believe that an approach based on statistical physics and stochastic dynamic systems theory provides a promising framework for describing and understanding learning dynamics in neural networks, which may also lead to more efficient learning algorithms.

1. Introduction: learning as a stochastic dynamical system

Modern artificial neural network (ANN)-based algorithms, in particular, deep-learning neural networks (DLNNs) [1, 2] have enjoyed a long string of tremendous successes, achieving human-level performance in image recognition [3], machine translation [4], games [5], and even solving long-standing grand-challenge scientific problems, such as protein folding [6]. However, despite DLNNs' successes, the underlying mechanism of how they work remains unclear. For example, one key ingredient in powerful DLNNs is a relatively simple iterative method called stochastic gradient descent (SGD) [7, 8]. However, the reason why SGD is so effective at finding highly generalizable solutions in high-dimensional nonconvex loss-function landscapes remains unclear. Random elements due to subsampling in SGD seem to be key for learning, yet the inherent noise in SGD also makes it difficult to understand.

From thermodynamics and statistical physics, we know that physical systems with many degrees of freedom are subject to stochastic fluctuations, e.g., thermal noise that drives Brownian motion, and powerful tools have been developed to understand collective behaviors in stochastic processes [9]. In this paper, we propose to consider the SGD-based learning process as a stochastic dynamic system and to investigate SGD-based learning dynamics using concepts and methods from statistical physics.

In an ANN, the model is parameterized by its weights, represented as an N_p-dimensional vector: $w = (w_1, w_2, ..., w_{N_p})$, where N_p is the number of parameters (weights). The dynamics of learning in ANN can thus be described by the motion of a "learner" particle (with coordinates w) in the weight space. Supervised learning uses a set of N training samples, each with an input vector X_k and a correct output vector Z_k for $k = 1, 2, ..., N$. For each input X_k, the learning system predicts an output vector $Y_k = G(X_k, w)$, where the output function G depends on the architecture of the NN as well as its weights, w. The goal of learning is to discover the weight parameters that minimize the difference between the predicted and correct output characterized by an overall loss function (or energy function):

$$L(w) = N^{-1} \sum_{k=1}^{N} l_k, \tag{1}$$

where $l_k = d(Y_k, Z_k)$ is the loss for sample k that measures the distance between Y_k and Z_k. A popular choice for d is the cross-entropy loss, which is what we use in this paper.

One learning strategy is to update the weights by following the gradient of L directly. However, this direct gradient descent (GD) scheme is computationally prohibitive for large datasets and it also has the obvious shortfall of being trapped by local minima or saddle points. SGD was first introduced to circumvent the large dataset problem by updating the weights according to a subset (minibatch) of samples randomly chosen at each iteration [7]. Specifically, the change of weight w_i ($i = 1, 2, \ldots, N_p$) for iteration t in SGD is given by

$$\Delta w_i(t) = -\alpha \frac{\partial L^{\mu(t)}(w)}{\partial w_i}, \tag{2}$$

where α is the learning rate and $\mu(t)$ represents the random minibatch used for iteration t. The mini loss function for a minibatch μ of size B is defined as follows:

$$L^{\mu}(w) = B^{-1} \sum_{l=1}^{B} d(Y_{\mu_l}, Z_{\mu_l}), \tag{3}$$

where μ_l ($l = 1, 2, \ldots, B$) labels the B randomly chosen training samples.

In addition to the computational advantage of SGD, the inherent noise due to random subsampling in SGD allows the system to escape local traps. In SGD, noise originates from the difference between the minibatch loss function L^{μ} and the whole-batch loss function, L: $\delta L^{\mu} \equiv L^{\mu} - L$. Using the continuous time approximation of equation (2), the SGD learning dynamics can be described by a Langevin equation:

$$\frac{dw}{dt} = -\alpha \nabla_w L + \eta, \tag{4}$$

where the first term on the right-hand side (RHS) of equation (4) is the usual deterministic GD term, and the second term corresponds to SGD noise, defined as: $\eta \equiv -\alpha \nabla \delta L^{\mu}$. The SGD noise has a zero mean $\langle \eta \rangle_{\mu} = 0$, and its strength is characterized by the noise matrix $\Delta_{ij} \equiv \langle \eta_i \eta_j \rangle = \alpha^2 C_{ij}$, where the covariance matrix $\mathbf{C}$ can be written as follows:

$$C_{ij} \equiv \left\langle \frac{\partial \delta L^{\mu}}{\partial w_i} \frac{\partial \delta L^{\mu}}{\partial w_j} \right\rangle_{\mu} = \left\langle \frac{\partial L^{\mu}}{\partial w_i} \frac{\partial L^{\mu}}{\partial w_j} \right\rangle_{\mu} - \frac{\partial L}{\partial w_i} \cdot \frac{\partial L}{\partial w_j}. \tag{5}$$

According to equation (4), the SGD-based learning dynamics can be considered as the stochastic motion of the learner particle in the high-dimensional weight space. The stochastic dynamics of physical systems that are in thermal equilibrium can also be described by Langevin equations with the same deterministic term as in equation (4), but with a much simpler noise term that describes the isotropic and homogeneous thermal fluctuations. Indeed, as first pointed out by Chaudhari and Soatto [10], SGD noise is neither isotropic nor homogeneous in the weight space. In this sense, SGD noise is highly nonequilibrium. As a result of nonequilibrium SGD noise, the steady-state distribution of weights is not the Boltzmann distribution seen in equilibrium systems, and the SGD dynamics exhibits much richer behavior than simply minimizing a global loss function (free energy).

How can we understand SGD-based learning in ANN? Here, we propose to bring useful concepts and tools from statistical physics [11] and stochastic processes [9] to bear on characterizing and investigating the SGD learning process/dynamics. In the rest of this paper, we describe a systematic way to characterize SGD dynamics based on order parameters that are defined over the minibatch gradient ensemble. We show how this approach allows us to identify and understand various phases of the learning process with and without labeling noise, which may lead to useful algorithms that improve generalization in the presence of mislabeled data. Throughout our study, we use realistic but simple datasets to demonstrate the principles of our approach, and pay less attention to absolute performance.

2. Characterizing SGD learning dynamics: the minibatch gradient ensemble and order parameters

To characterize the stochastic learning dynamics in SGD, we introduce the concept of a minibatch ensemble $\{\mu\}$, where each member of the ensemble is a minibatch with B samples chosen randomly from the whole training dataset (of size N). Based on the minibatch ensemble, we can define an ensemble of minibatch loss functions L^μ or, equivalently, an ensemble of gradients $\{g^\mu(\equiv -\nabla L^\mu(w))\}$ at each weight vector w.

The SGD learning dynamics is fully characterized by the statistical properties of the gradient ensemble in weight space $\{g^\mu(w)\}$. At each point in the weight space, the ensemble average of the minibatch gradients is the gradient over the whole dataset: $g(w) \equiv \langle g^\mu(w)\rangle_\mu (= \nabla L(w))$, and fluctuations of the gradients around their mean give rise to the noise matrix (equation (5)). To measure the alignment between the minibatch gradients, we define an alignment parameter R:

$$R(w) \equiv \langle \hat{g}^\mu(w) \cdot \hat{g}^\nu(w)\rangle_{\mu,\nu}, \tag{6}$$

where $\hat{g}^\mu = g^\mu/\|g^\mu\|$ is the unit vector in the gradient direction g^μ. The alignment parameter is the cosine of the relative angle between the two gradients averaged over all pairs of minibatches (μ, ν) in the ensemble.

To analyze the gradient fluctuations in different directions, we can project the minibatch gradient g^μ onto the mean, g, and write it as follows:

$$g^\mu = g^\mu_\perp + \lambda_\mu g, \tag{7}$$

where $\lambda_\mu = (g^\mu \cdot g)/\|g\|^2$ is the projection constant and $g^\mu_\perp$ is the residue gradient perpendicular to g: $g^\mu_\perp \cdot g = 0$. Analogously to kinetic energy, we use the square of the gradient to measure the learning activity. The ensemble averaged activity (A) can be split into two parts:

$$A \equiv \langle \|g^\mu\|^2\rangle_\mu = \langle \|g^\mu_\perp\|^2\rangle_\mu + \langle \lambda^2_\mu\rangle_\mu \|g\|^2 \equiv A_\perp + A_\parallel, \tag{8}$$

where $A_\parallel$ and $A_\perp$ represent activities along the mean gradient and orthogonal to it, respectively.

The total variance, D, of fluctuations in all directions is the trace of the covariance matrix $\mathbf{C}$:

$$D \equiv Tr(\mathbf{C}) = \sum_i C_{ii} = A_\perp + D_\parallel, \tag{9}$$

where $D_\parallel = \sigma^2_\lambda \|g\|^2$ is the variance along the direction of the batch gradient g and $\sigma^2_\lambda \equiv \langle \lambda^2_\mu\rangle_\mu - 1$ is the variance of λ_μ (Note that $\langle \lambda_\mu\rangle_\mu = 1$ by definition); $A_\perp$ is the total variance in the orthogonal directions. The mean learning activity can be written as: $A = A_0 + A_\perp + D_\parallel$, where $A_0 \equiv \|g\|^2$ represents the directed activity in the direction of the mean gradient; $A_\perp$ and $D_\parallel$ represent the diffusive search activities in the directions orthogonal and parallel to the mean gradient, respectively.

All these quantities (A, A_0, R, σ^2_λ) depend on the weights (w). Along an SGD learning trajectory in weight space, we can evaluate these order parameters and their relative values at any given time t to characterize different phases of the SGD learning dynamics. For example, we use A and A_0 to measure the total learning activity and the activity in the mean gradient direction, respectively. The alignment between different minibatch gradients is measured by R, which is related to the fractionally aligned activity A_0/A. The fluctuations of the minibatch gradients projected onto the mean gradient are measured by σ^2_λ. In our previous work [12], we used time averaging to approximate some of these order parameters for computational convenience. However, the properties of the SGD dynamics at any given point in weight space are precisely defined by these ensemble-averaged order parameters, and are used hereafter.

As previously mentioned, SGD noise is anisotropic and varies in weight space. The positive-definite eigenvalue e_l of the symmetric covariance matrix $\mathbf{C}$ is the noise strength in the corresponding eigen-direction ($l = 1, 2, \ldots, N_p$, where N_p is the number of weights or the dimensions of the weight space). The overall noise strength $D = Tr(\mathbf{C}) = \sum_{l=1}^{N_p} e_l$ describes the total search activity, and the eigenvalue spectrum

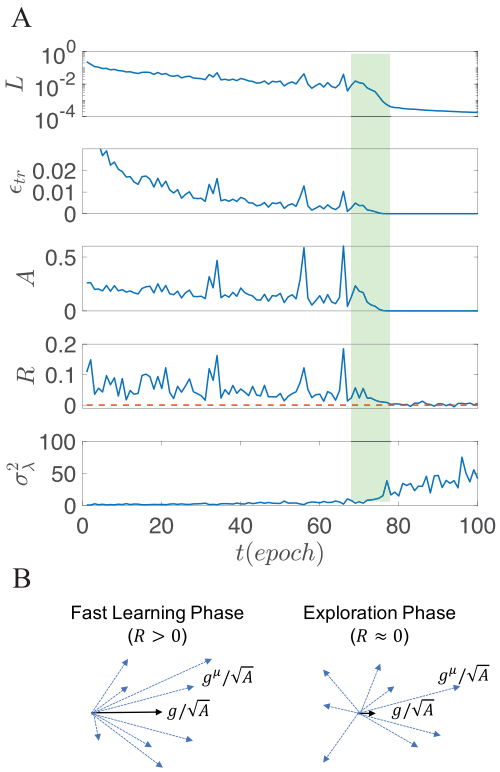

Figure 1. Two phases of learning without labeling noise. (A) Training loss L, training error ϵ_{tr}, and order parameters A, R, and σ^2_λ versus (training) time. The fast learning phase corresponds to a directed (finite $R > 0$, $\sigma^2_\lambda \sim 1$) and fast (large A) motion in weight space; the exploration phase corresponds to a diffusive ($R \approx 0$, $\sigma^2_\lambda \gg 1$) and slow (small A) motion in weight space. The dotted line shows $R = 0$. The green bar highlights the transition region. The MNIST data and a fully connected network with two hidden layers (30×30) are used here. (B) Illustration of the normalized minibatch gradient ensemble (blue dotted arrows) and their means (black solid arrows) in the two learning phases.

$\{e_l,\ l = 1, 2, \ldots, N_p\}$ tells us how much of the total search activity is spent in each eigen-direction. From the noise spectrum, we can define the effective dimension of the search activity $D_s(w)$ as the number of dimensions wherein the variance in the subspace of parameters accounts for a certain large percentage (e.g. 90%) of the total variance D.

3. Phases of SGD learning dynamics in the absence of mislabeled data

We first study the learning dynamics without mislabeled data, e.g., the original MNIST dataset (details of all numerical experiments can be found in the supplemental material (available online at stacks.iop.org/MLST/2/043001/mmedia)). As shown in figure 1, the dynamics of the overall loss function L suggests that there are two phases in learning. There is an initial fast learning phase, where L decreases quickly, followed by an exploration phase where the training error ϵ_{tr} reaches zero (or nearly zero), while L still decreases, but much more slowly. These two learning phases exist independently of hyperparameters (e.g. α and B) and network architectures (all connected networks or CNNs) used for different datasets (e.g., MNIST and CIFAR). The weights reached in the exploration phase can be considered as solutions to the problem, given that the training error vanishes.

The dynamics of the order parameters $A(t)$, $R(t)$, and σ^2_λ along the trajectory can be used to characterize and understand the two phases. As shown in figure 1(A), at the beginning of the learning process, the learning activity A is relatively large, and the alignment parameter R is finite. In this initial phase of learning, the minibatch gradients have a high degree of alignment, resulting in a strongly directed motion of the learner particle and a rapid decrease of L toward a solution region in the weight space with low L and zero training error ϵ_{tr}. In the exploration phase, the average learning activity A becomes much smaller, while the average alignment parameter R approaches zero. This means that the motion of the weighted particle becomes mostly diffusive (weakly directed) and the decrease of L slows. This diffusive motion of the weights allows the system to explore the solution space. The transition from a directed motion to a diffusive motion is also reflected in the large increase in the variance σ^2_λ at the transition. Due to the finite size of the system, the transition is not infinitely sharp, like the phase transitions that occur in physical systems in the

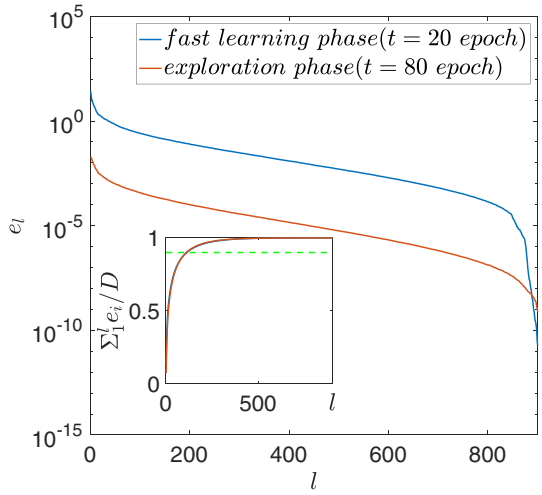

Figure 2. The noise spectra, i.e. rank-ordered eigenvalues $\{e_l,\ l = 1, 2..., N_p\}$ in the fast learning phase (black) and the exploration phase (red). The inset shows the normalized accumulated variance $D^{-1}\sum_{i=1}^{l} e_i$. The two spectra are similar, except for their total variance, D. The effective dimension $D_s \sim 110$, which is much smaller than the number of parameters ($N_p = 900$), is roughly the same in both phases. The data and network used here are the same as in figure 1.

thermodynamic limit (infinite system limit). As shown in figure 1(A), the training error ϵ_{tr} becomes zero during the transition regime and it stays at zero during the exploration phase. These results confirm the results of our previous study, which used time-averaged ordered parameters [12]. The key differences between the two phases in terms of the alignment of minibatch gradients and the mean gradient strength are illustrated in figure 1(B). These two phases are independent of the network size, and they also appear in other neural network architectures, such as convolutional neural networks and residual networks. See figure S1 in the supplementary material for details.

We have also studied the noise spectra in the two phases. As shown in figure 2, unlike isotropic thermal noise, SGD noise has a highly anisotropic structure with most of its variance (strength) concentrated in a relatively small number of directions. The normalized noise spectra are similar in both phases and the total noise strength (variance) D is much higher in the fast learning phase. The effective dimension, defined as the number of directions that contains 90% of the total variance, is $D_s \sim 110$, which is much smaller than the number of weights (parameters), and remains roughly constant as the number of parameters increases.

4. Phases of SGD learning dynamics in the presence of mislabeled data

There has been much interest in deep learning in the presence of mislabeled data. This was triggered by a recent study [13], in which the authors showed that random labels can easily be fitted by deep networks in the over-parameterized regime and that such overfitting destroys generalization. Here, we report some new results using the dynamic systems approach developed in the previous sections to study SGD learning dynamics with labeling noise.

In a dataset with N_c correctly labeled training samples and N_w incorrectly (randomly) labeled samples, the overall loss function L consists of two parts, L_c and L_w, which originate from the correctly labeled samples and the randomly labeled samples, respectively:

$$L = (1-\rho)L_c + \rho L_w = N^{-1}\left[\sum_{k=1}^{N_c} l_k + \sum_{k=1}^{N_w} \tilde{l}_k\right], \tag{10}$$

where $N = N_c + N_w$ is the total number of training samples and $\rho = N_w/N$ is the fraction of mislabeled samples. The loss function for a correctly labeled sample is the cross-entropy l between the output $Y_k(X_k, w)$ of the network with weight vector w and the correct label vector Z_k: $l_k = l(Y_k, Z_k)$, while the loss function for a mislabeled sample is: $\tilde{l}_k = l(Y_k, Z_k^r)$, where Z_k^r is a random label vector.

We conducted experiments using MNIST and CIFAR10 with different fractions of mislabeled data (ρ). As shown in figure 3(A) for MNIST, the whole learning process can be divided into four phases (the study of the CIFAR10 dataset showed similar results):

- Phase I: During this initial fast learning phase (0–10 epochs in figure 3(A)), the test error ϵ_{te} decreases quickly as the system learns the correctly labeled data. The error ϵ_c from the correctly labeled training data follows

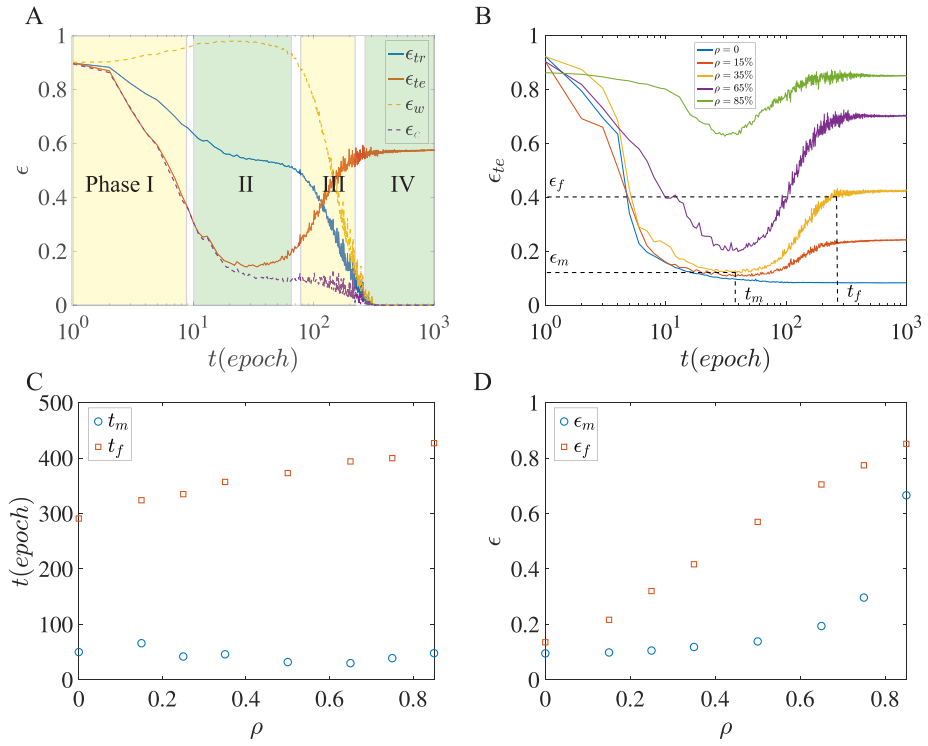

Figure 3. Learning dynamics in the presence of labeling noise. (A) The training error ϵ_{tr}, the test error ϵ_{te}, the training error for correctly labeled data ϵ_c, and the training error for mislabeled data ϵ_w are shown for a subset of MNIST data with 400 samples per digit and a fully connected network with two hidden layers (50 hidden units per layer). SGD hyper-parameters: $B = 25$, $\alpha = 0.01$. (B) ϵ_{te} dynamics for different values of ρ. (C) The dependence of the time scales (t_m and t_f) on ρ. (D) The dependence of the minimum and final test errors (ϵ_m and ϵ_f) on ρ.

the exact same trend as ϵ_{te}, and the error ϵ_w from the mislabeled training data actually increases slightly, indicating that the learning in phase I is dominated by the correctly labeled training data.

- Phase II: After the initial fast learning phase, the test error ϵ_{te} stays roughly the same during phase II (10–70 epochs in figure 3(A)). Both ϵ_w and ϵ_c remain flat, indicating that learning activities for the correct and incorrect samples are balanced during phase II. This can also be seen in the plateau in the total training error $\epsilon_{tr} = (1-\rho)\epsilon_c + \rho\epsilon_w$.
- Phase III: At the end of phase II (∼70 epochs), the test error ϵ_{te} starts to increase quickly, while the training errors for both the correct and the incorrect training data (ϵ_c, ϵ_w) decrease to zero during phase III (70–200 epochs). During phase III, the system finally manages to find (learn) a solution that satisfies both the correct and incorrect training data.
- Phase IV: Phase IV corresponds to the slow exploration phase after the system reaches the solution space for the whole dataset. The test error reaches a high plateau in phase IV.

The four distinct phases in the presence of labeling noise, and the corresponding 'U'-shaped behavior of the test error, are general for a wide range of noise levels (ρ), see figure 3(B). Quantitatively, the dynamics of the test error $\epsilon_{te}(t)$ during these four phases can be characterized by two timescales: t_m—the time when the test error reaches its minimum and t_f—the time when the training loss function reaches its minimum, and the two corresponding test errors: ϵ_m and ϵ_f. All four parameters depend on ρ. As shown in figure 3(C), t_m is almost independent of ρ, which means that learning the correctly labeled data is independent of the data size, as long as the data size is large enough. However, t_f increases with ρ, which means that the network needs more time to memorize the incorrectly labeled data as the number of mislabeled samples increases. As shown in figure 3(D), the final test error ϵ_f increases almost linearly with ρ, which is caused by the increased fraction of mislabeled data. The minimum error ϵ_m remains roughly the same when ρ is small, but increases sharply after a threshold and approaches ϵ_f when $\rho > 0.85$. This also makes sense, because when ρ is large, learning is dominated by mislabeled data and the correctly labeled data no longer drives the learning dynamics.

Here, we try to understand the different phases and the transitions between them by using order parameters that are modified for the case with labeling noise. In particular, each minibatch μ now consists of two smaller minibatches, μ_c and μ_w, for correctly and incorrectly labeled data ($\mu = \mu_c + \mu_w$) with average

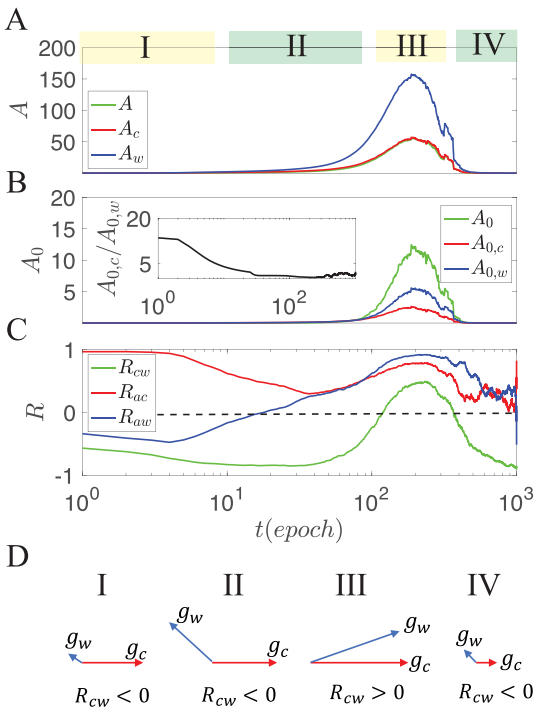

Figure 4. Dynamics of the order parameters during phases of learning with mislabeled data. (A) Total activities (A, A_w, A_c). (B) Directed activities (A_0, $A_{0,w}$, $A_{0,c}$), the inset shows the ratio $A_{0,c}/A_{0,w}$. (C) Alignment parameters (R_{cw}, R_{ac}, R_{aw}). The dotted line shows $R = 0$. (D) Illustration of the four different phases in terms of the relative strength and direction of the two mean gradients (g_c and g_w). $\rho = 0.5$ is used for (A)–(C).

sizes of $B_c = (1-\rho)B$ and $B_w = \rho B$, respectively. The minbatch loss function can be decomposed into two minibatch loss functions, L^{μ_c} and L^{μ_w}, defined separatelyfor μ_c and μ_w : $L^{\mu} = L^{\mu_c} + L^{\mu_w}$. At a given point in weight space, the ensemble-averaged gradient and activity for the correctly and incorrectly labeled data can be defined separately:

$$g_c \equiv \left\langle \frac{\partial L^{\mu_c}}{\partial w} \right\rangle_{\mu_c} = \frac{\partial L_c}{\partial w}, \quad A_c \equiv \left\langle \left\| \frac{\partial L^{\mu_c}}{\partial w} \right\|^2 \right\rangle_{\mu_c}, \tag{11}$$

$$g_w \equiv \left\langle \frac{\partial L^{\mu_w}}{\partial w} \right\rangle_{\mu_w} = \frac{\partial L_w}{\partial w}, \quad A_w \equiv \left\langle \left\| \frac{\partial L^{\mu_w}}{\partial w} \right\|^2 \right\rangle_{\mu_w}. \tag{12}$$

The alignment of the two gradients g_c and g_w can be characterized by the cosine of their relative angle:

$$R_{cw} \equiv \frac{g_c \cdot g_w}{\|g_c\| \|g_w\|}, \tag{13}$$

from which we obtain the ensemble-averaged gradient and activity for the whole dataset:

$$g \equiv \left\langle \frac{\partial L^{\mu}}{\partial w} \right\rangle_{\mu} = (1-\rho) g_c + \rho g_w, \tag{14}$$

$$A \equiv \left\langle \left\| \frac{\partial L^{\mu}}{\partial w} \right\|^2 \right\rangle_{\mu} = (1-\rho)^2 A_c + \rho^2 A_w + 2\rho(1-\rho)\|g_c\|\|g_w\| C_{cw}. \tag{15}$$

From the basic ordered parameters defined above, we can define the directed activity $A_{0,c} \equiv (1-\rho)^2 \|g_c\|^2$, $A_{0,w} \equiv \rho^2 \|g_w\|^2$, and $A_0 \equiv \|g\|^2 = A_{0,c} + A_{0,w} + 2[A_{0,w}A_{0,c}]^{\frac{1}{2}} C_{cw}$; and the alignments between g and g_c, and between g and g_w are: $R_{aw} \equiv \frac{g \cdot g_w}{\|g\|\|g_w\|}$, $R_{ac} \equiv \frac{g \cdot g_c}{\|g\|\|g_c\|}$. We can also define alignment order parameters among members within the different gradient ensembles ($\{\mu_c\}$, $\{\mu_w\}$, and $\{\mu\}$).

We studied three groups of order parameters: the total activities (A, A_c, A_w); the directed activities (A_0, $A_{0,c}$, $A_{0,w}$) and their alignments (R_{cw}, R_{aw}, R_{ac}) to understand the learning dynamics in the presence of labeling noise. In figure 4, we show how these order parameters change during training for the case with

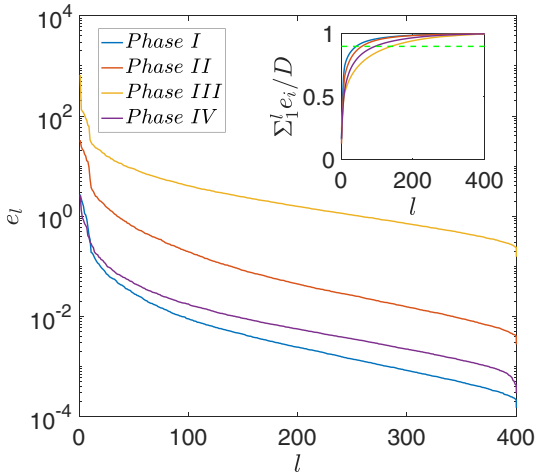

Figure 5. The noise spectra, i.e. rank-ordered eigenvalues $\{e_l,\ l = 1, 2..., N_p\}$ in different phases of learning with labeling noise (using the tsame settings as in figure 4). The inset shows the normalized accumulated variance $D^{-1}\sum_{i=1}^{l} e_i$. The spectra are similar, except for their total variance D. In the different phases, the effective dimension D_s varies in the range of 50–150, which is much smaller than the number of parameters ($N_p = 2500$).

$\rho = 50\%$. As shown in figures 4(A) and (B), all the learning activity order parameters (A's and A_0's) show a consistent trend of increasing during phases I, II, and III before decreasing during phase IV. This is in contrast to the behavior of learning activity A in the absence of labeling noise, which shows a relatively flat or slightly decreasing trend during the fast learning phase (see figure 1). This continuously elevated learning activity in phases I–III suggests an increasing frustration between the two separate learning tasks (of learning the correctly and incorrectly labeled datasets) before a consistent solution is found in phase IV.

The difference between the learning phases I, II, and III can be understood by studying the relation between the two mean gradients g_w and g_c characterized by the alignment order parameter R_{cw} (see figure 4(C)) and the relative strength of the two directed activities $A_{0,c}$ and $A_{0,w}$.

- Phase I: $A_{0,c} \gg A_{0,w}$, $R_{cw} < 0$. In phase I, the directed activity from the correctly labeled data is much larger than that from the incorrectly labeled data (see inset in figure 4(B)). This is due to the fact that samples from the correctly labeled dataset are consistent with each other in terms of their labels, which leads to a much larger mean gradient toward learning a solution for the correctly labeled data. In phase I, g_c and g_w are not aligned ($R_{cw} < 0$). Due to the fact that $A_{0,c} \gg A_{0,w}$, we have $R_{aw} < 0$, which means that there is an increase of L_w during phase I, as observed in figure 3(A).
- Phase II: $A_{0,w} \approx A_{0,c}$, $R_{cw} < 0$. As the system approaches a solution for the correctly labeled data during the late stage of phase I, the directed learning activity from the mislabeled data ($A_{0,w}$) increases sharply, and $A_{0,w}$ becomes comparable with $A_{0,c}$ in phase II (see the inset of the middle panel in figure 4). In addition, the two mean gradients (g_c and g_w) are opposite to each other, with $R_{cw} \approx -1$. As a result of the balanced gradients between the two datasets, the overall directed activity is small $A_0 \ll A_{0,c(w)}$ and the loss functions (L_c, L_w, and L) remain relatively flat during phase II (see figure 3(A)).
- Phase III: $A_{0,w} \approx A_{0,c}$, $R_{cw} > 0$. The system enters phase III when it finally finds a direction that decreases both loss functions (L_w and L_c) as evidenced by the alignment of g_c and g_w, which only happens during phase III. This alignment ($R_{cw} > 0$) means that the system can finally learn a solution for all the training data.
- Phase IV: $A_{0,w} \approx A_{0,c}$, $R_{cw} < 0$. Once the system finds a solution for all data, learning slows down to explore other solutions nearby. Phase IV is similar to the exploration phase without mislabeled data, where learning activity is much reduced compared to that of phases I–III.

The key differences between the four phases, in terms of the strength and relative direction of the two mean gradients (g_c and g_w), are illustrated in figure 4(D).

We have also analyzed the noise spectra in the different learning phases in the presence of labeling noise. As shown in figure 5, the normalized spectra remain roughly the same in different learning phases and the effective dimensions are $D_{I,II,III,IV} \approx 43, 58, 140, 95$, which are much smaller than the number of parameters. We note that both the noise spectra and the effective noise dimensions are similar to those without labeling noise (figure 2).

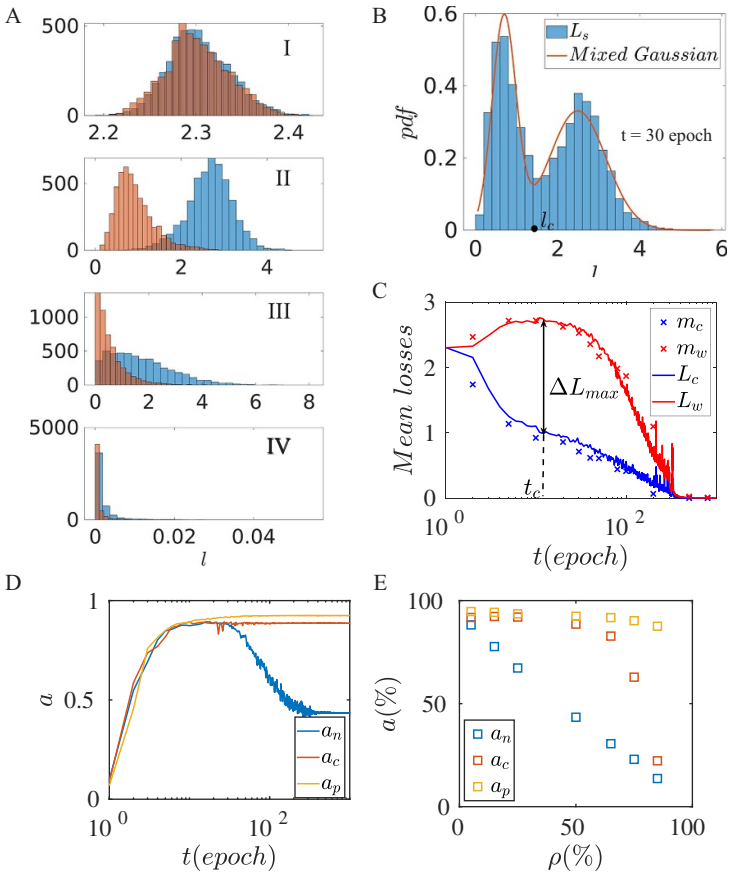

Figure 6. The individual loss distribution and the cleansing method. (A) The loss distributions of correctly labeled samples (red) and mislabeled samples (blue) in different learning phases. (B) The bimodal distribution in phase II can be fitted by a Gaussian mixture model (red line), which is used to determine a threshold l_c for cleansing. (C) The mean losses (symbols) predicted by the Gaussian mixture model agree with their true values from experiments (lines). A cleansing time t_c can be determined when $\Delta L(\equiv m_w - m_c)$ reaches its maximum. (D) The test accuracy without cleansing (a_n), with cleansing (a_c), and with only the correctly labeled training data (a_p) versus training time. The labeling noise level $\rho = 50\%$ for (A)–(D). (E) a_n, a_c, and a_p versus ρ. The slight decrease in a_p as ρ increases is due to the decreasing size of the correctly labeled dataset. The MNIST dataset and network used here are the same as those in figure 3.

5. Identifying and cleansing the mislabeled samples in phase II

Our study so far has used various ensemble-averaged properties to demonstrate the different phases of learning dynamics. We now investigate the distribution of losses for individual samples and how the individual loss distribution evolves with time. In figure 6(A), we show the probability distribution functions (PDFs)—$P_c(l, t)$ and $P_w(l, t)$—for the individual losses of the correctly and incorrectly labeled samples at different times during training. Starting with an identical distribution at time zero, the two distributions quickly separate during phase I as $P_c(l, t)$ moves to smaller losses while $P_w(l, t)$ moves to slightly higher losses. The separation between the two distributions increases during phase I and reaches its maximum during phase II. After the system enters phase III, the gap between the two distributions closes quickly as the system learns the mislabeled data and $P_w(l, t)$ catches up with $P_c(l, t)$ at small losses. In phase IV, these two distributions become indistinguishable again as they both become highly concentrated at near-zero losses.

As a result of the different dynamics of the two distributions, the overall individual loss distribution $P(l) = (1 - \rho)P_c(l) + \rho P_w(l)$ exhibits a bimodal behavior, which is most pronounced during phase II. We can fit the overall distribution using a Gaussian mixture model: $l \sim (1 - r)\mathcal{N}(m_c, s_c^2) + r\mathcal{N}(m_w, s_w^2)$, with the following fitting parameters: fraction r, means $m_{c,w}$, and variances $s_{c,w}^2$. As shown in figure 6(B), the Gaussian mixture model fits $P(l)$ well, and furthermore, the fitted means m_c and m_w agree with the mean losses (L_c, and L_w) obtained from experiments.

The separation of individual loss distribution functions has recently been used to devise sophisticated methods to improve generalization, such as those reported in [14, 15]. Here, we demonstrate the basic idea by presenting a simple method to identify and clean the mislabeled samples based on the understanding of different learning phases. In particular, according to our analysis, such a cleansing process can be best done during phase II. For simplicity, we set the time t_c for cleansing to be when the difference $\Delta L(\equiv m_w - m_c)$

reaches its maximum. At $t = t_c$, we can set a threshold l_c, which best separates the two distributions. For example, we can set l_c as the loss when the two PDFs are equal or simply as the average of m_c and m_w (we do not observe significant differences between the two choices). We can then remove all the data that have a loss larger than l_c and continue training with the cleansed dataset. Alternatively, we can stop the training altogether at $t = t_c$, i.e. early stopping. In our experiments, we did not observe significant differences between these two choices. In figure 6(D), the test accuracies a_n (without cleansing), a_c (with cleansing), and a_p (with only the correctly labeled data) are shown for MNIST data with $\rho = 50\%$ labeling noise. The performance of the cleansing algorithm can be measured by $Q = \frac{a_c - a_n}{a_p - a_n}$, which depends on the noise level ρ. As shown in figure 6(E), the cleansing method can achieve a significant improvement in generalization ($Q > 50\%$) for noise levels of up to $\rho = 80\%$ noise level. The details of the data cleansing procedure are described in the supplementary materials.

6. Summary

DLNNs have demonstrated tremendous capability for learning and problem solving in diverse domains. However, the mechanism underlying this seemingly magical learning ability is not well understood. For example, modern DNNs often contain more parameters than training samples, which allow them to interpolate (memorize) all the training samples, even if their labels are replaced by pure noise [16, 17]. Remarkably, despite their huge capacity, DNNs can achieve small generalization errors on real data (this phenomenon has been formalized as the so-called 'double descent' curve [18–23]). The learning system/model seems to be able to self-tune its complexity in accordance with the data to find the simplest possible solution in a highly over-parameterized weight space. However, the way in which the system adjusts its complexity dynamically, and how SGD seeks out simple and more generalizable solutions for realistic learning tasks remain poorly understood.

In this paper, we demonstrate that our approach based on statistical physics and stochastic dynamical systems provides a useful theoretical framework (an alternative to the traditional theorem-proving approach) for studying SGD-based machine learning by applying it to the identification and characterization of the different phases of SGD-based learning, with and without labeling noise. In an earlier work [12], we used this approach to study the relation between SGD dynamics and the loss function landscape, and discovered an inverse relation between weight variance and the loss landscape flatness that is the opposite of the fluctuation–dissipation relation (akin to the Einstein relation) in equilibrium systems. We believe this framework may pave the way for a deeper understanding of deep learning by bringing powerful ideas (e.g., phase transitions in critical phenomena) and tools (e.g., renormalization group theory and replica methods) from statistical physics to bear on understanding ANN. It would be interesting to use this general framework to address other fundamental questions in machine learning, such as generalization [24–26] (in particular, the mechanism for the double descent behavior in learning as described above), the relation between task complexity and network architecture, and information flow in DNNs [27, 28], as well as building a solid theoretical foundation for important applications, such as transfer learning [29], curriculum learning [30], and continuous learning [31–33].

Acknowledgments

We thank Mark Wegman, Haifeng Qian and Tom Theis for discussions. The work by Y.F. was done when he was an IBM intern.

Data availability statement

The data that support the findings of this study are available upon reasonable request from the authors.

ORCID iDs

Yu Feng https://orcid.org/0000-0003-3688-0284
Yuhai Tu https://orcid.org/0000-0002-4589-981X

References

[1] LeCun Y, Bengio Y and Hinton G 2015 Deep learning *Nature* **521** 436 E
[2] Goodfellow I, Bengio Y, Courville A and Bengio Y 2016 *Deep Learning* vol 1 (Cambridge, MA: MIT Press)
[3] He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition *Proc. IEEE Conference on Computer Vision and Pattern Recognition* pp 770–8

[4] Wu Y *et al* 2016 Google's neural machine translation system: bridging the gap between human and machine translation (arXiv:1609.08144)
[5] Silver D *et al* 2016 Mastering the game of go with deep neural networks and tree search *Nature* **529** 484–9
[6] Callaway E 2020 'It will change everything': Deepmind's ai makes gigantic leap in solving protein structures *Nature* **588** 203–4
[7] Robbins H and Monro S 1951 A stochastic approximation method *Ann. Math. Stat.* **22** 400–7
[8] Bottou L Lechevallier Y and Saporta G 2010 Large-scale machine learning with stochastic gradient descent *Proc. COMPSTAT'2010*, ed Y Lechevallier G Saporta (Heidelberg: Physica-Verlag HD) pp 177–86
[9] Kampen N G V 2010 *Stochastic Processes in Physics and Chemistry* (Amsterdam: Elsevier)
[10] Chaudhari P and Soatto S 2018 Stochastic gradient descent performs variational inference, converges to limit cycles for deep networks *2018 Information Theory and Applications Workshop (ITA)*
[11] Forster D 2018 *Hydrodynamic Fluctuations, Broken Symmetry and Correlation Functions* (Boca Raton, FL: CRC Press)
[12] Feng Y and Tu Y 2021 The inverse variance–flatness relation in stochastic gradient descent is critical for finding flat minima *Proc. Natl Acad. Sci.* **118** e2015617118
[13] Zhang Y, Saxe A M, Advani M S and Lee A A 2018 Energy–entropy competition and the effectiveness of stochastic gradient descent in machine learning *Mol. Phys.* **116** 3214–23
[14] Arazo E, Ortego D, Albert P, O'Connor N E and McGuinness K 2019 Unsupervised label noise modeling and loss correction (arXiv:1904.11238)
[15] Li M, Soltanolkotabi M and Oymak S 2020 Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks (arXiv:1903.11680)
[16] Zhang C, Bengio S, Hardt M, Recht B and Vinyals O 2016 Understanding deep learning requires rethinking generalization (arXiv:1611.03530)
[17] Arpit D *et al* 2017 A closer look at memorization in deep networks (arXiv:1706.05394)
[18] Belkin M, Hsu D, Ma S and Mandal S 2019 Reconciling modern machine-learning practice and the classical bias–variance trade-off *Proc. Natl Acad. Sci.* **116** 15849–54
[19] Brutzkus A, Globerson A, Malach E and Shalev-Shwartz S 2017 SGD learns over-parameterized networks that provably generalize on linearly separable data (arXiv:1710.10174)
[20] Li Y and Liang Y 2018 Learning overparameterized neural networks via stochastic gradient descent on structured data *Adv. Neural Inf. Process. Syst.* **31** 8157–66
[21] Mei S and Montanari A 2019 The generalization error of random features regression: precise asymptotics and double descent curve (arXiv:1908.05355)
[22] Geiger M *et al* 2020 Scaling description of generalization with number of parameters in deep learning *J. Stat. Mech.: Theory Exp.* **2020** 023401
[23] Gerace F, Loureiro B, Krzakala F, Mézard M and Zdeborová L 2020 Generalisation error in learning with random features and the hidden manifold model (arXiv:2002.09339)
[24] Neyshabur B, Bhojanapalli S, McAllester D and Srebro N 2017 Exploring generalization in deep learning *NIPS*
[25] Advani M S and Saxe A M 2017 High-dimensional dynamics of generalization error in neural networks (arXiv:1710.03667)
[26] Jiang Y, Neyshabur B, Mobahi H, Krishnan D and Bengio S 2019 Fantastic generalization measures and where to find them (arXiv:1912.02178)
[27] Shwartz-Ziv R and Tishby N 2017 Opening the black box of deep neural networks via information (arXiv:1703.00810)
[28] Tishby N and Zaslavsky N 2015 Deep learning and the information bottleneck principle *2015 IEEE Information Theory Workshop (ITW)*
[29] Yosinski J, Clune J, Bengio Y and Lipson H 2014 How transferable are features in deep neural networks? (arXiv:1411.1792)
[30] Bengio Y, Louradour J, Collobert R and Weston J 2009 Curriculum learning *Proc. 26th Annual Int. Conf. on Machine Learning* pp 41–8
[31] Ring M B 1994 Continual learning in reinforcement environments *PhD Thesis* University of Texas at Austin, Austin, TX
[32] Lopez-Paz D and Ranzato M 2017 Gradient episodic memory for continuum learning *NIPS*
[33] Riemer M *et al* 2018 Learning to learn without forgetting by maximizing transfer and minimizing interference (arXiv:1810.11910)