

Spring-Block Theory of Feature Learning in Deep Neural Networks

Cheng Shi¹, Liming Pan², and Ivan Dokmanić^{1,3,*}

¹*Departement Mathematik und Informatik, University of Basel, Spiegelgasse 1, 4051 Basel, Switzerland*

²*School of Cyber Science and Technology, University of Science and Technology of China, 230026 Hefei, China*

³*Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, 306 North Wright Street, Urbana, Illinois 61801, USA*



(Received 25 October 2024; accepted 29 May 2025; published 27 June 2025)

Feature-learning deep nets progressively collapse data to a regular low-dimensional geometry. How this emerges from the collective action of nonlinearity, noise, learning rate, and other factors, has eluded first-principles theories built from microscopic neuronal dynamics. We exhibit a noise–nonlinearity phase diagram that identifies regimes where shallow or deep layers learn more effectively and propose a macroscopic mechanical theory that reproduces the diagram and links feature learning across layers to generalization.

DOI: [10.1103/ys4n-2tj3](https://doi.org/10.1103/ys4n-2tj3)

Deep neural networks (DNNs) progressively compute features from which the final layer generates predictions. When optimized via stochastic dynamics over a data-dependent energy, each layer learns to compute better features than the previous one [1], ultimately transforming the data to a regular low-dimensional geometry [2–7]. Feature learning is a striking departure from kernel machines or random feature models (RFM) which compute linear functions of *fixed* features [8–10]. How it emerges from microscopic interactions between millions of artificial neurons is a central open question in deep learning [11–15].

Even with a single hidden layer [16–19], the interplay between initialization [20], width [21–23], learning rate [24,25], batch size [26,27], and data [28–30] results in a bewildering range of training dynamics. Deeper networks can be analyzed in various asymptotic regimes [31,32], with simplified training [33,34], or without nonlinearity [35,36]. There have been exciting advances in the infinite-width limit [37,38] and on low-dimensional stochastic gradient descent (SGD) dynamics [39,40]. These give invaluable insight, but the full deep, nonlinear setting has eluded a statistical mechanics approach where features emerge from microscopic interactions. A change of perspective may help close this gap.

In this Letter, we take a thermodynamical, top-down approach and look for a simple phenomenological model which captures the feature learning phenomenology. We

show that DNNs can be mapped to a phase diagram defined by noise and nonlinearity, with phases where layers learn features uniformly, and where deep or shallow layers learn better. “Better” is quantified through *data separation*—the ratio of feature variance within and across classes. To explain this phase diagram, we propose a macroscopic theory of feature learning in deep, nonlinear neural networks: we show that the stochastic dynamics of a nonlinear spring-block chain with asymmetric friction fully reproduce the phenomenology of data separation over training epochs and layers. The phase diagram is universal with respect to the source of stochasticity: varying dropout rate, batch size, label noise, or learning rate all result in the same phenomenology.

Our findings generalize recent work showing that in many DNNs each layer improves data separation by the same factor, with surprising regularity [3]. This *law of data separation* can be proved for linear DNNs with a particular choice of data and initialization [41,42]. It is, however, puzzling why just as many networks do not abide by it. Figure 4 shows that training the same DNN with three different parameter sets results in strikingly different distributions of data separation over layers. Even linear DNNs induce a complex energy landscape [17,43–48] and nonlinear training dynamics [36] that can result in noneven separation. Understanding why this happens and how it affects generalization is key to understanding feature learning.

In our theory, spring elongations model data separation. The empirical risk exerts a load on the network to which its layers respond by separating the data, subject to nonlinearity modeled by friction. The difference in length between consecutive springs results in a load on the incident block. Friction models dynamical nonlinearity which absorbs load (or spoils the signal in gradients),

*Contact author: ivan.dokmanic@unibas.ch

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

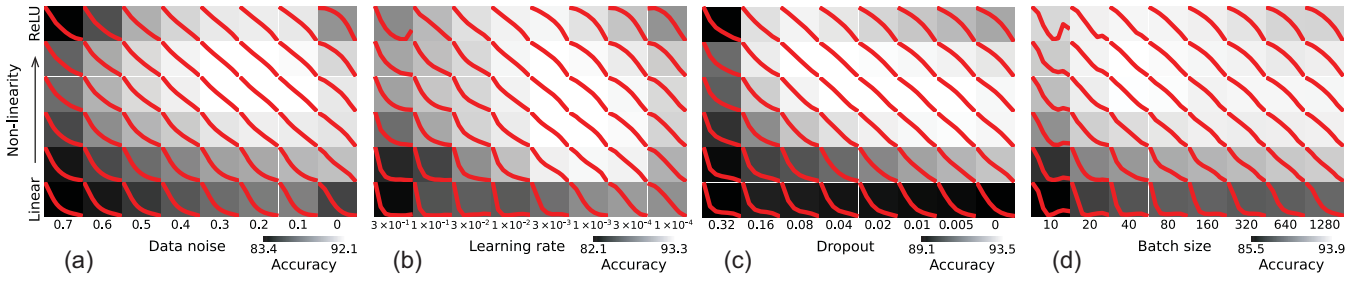


FIG. 1. Phase diagrams of DNN training load curves (red) for nonlinearity vs (a) data noise, (b) learning rate, (c) dropout, and (d) batch size. The nonlinearity is controlled by the negative slope in leaky ReLU, with values of 1, 0.8, 0.6, 0.4, 0.2, and 0 from the bottom row to the top row. In all cases, noise is strongest on the left, and nonlinearity strongest at the top. Background shading encodes test accuracy. Results are averaged over ten independent runs on MNIST.

causing the shallow layers to “extend” (separate data) less. Stochasticity from SGD [49,50], dropout [51], or noisy data [52], reequilibrates the load.

The resulting model reproduces the dynamics and the phase diagram of feature learning surprisingly well. It explains when data separation is uniformly distributed across layers and when deep or shallow layers learn faster. It shows why depth may hurt and why nonlinearity is a double-edged sword, resulting in expressive models but facilitating overfitting. A stability argument suggests a link between generalization and layerwise data separation which we remarkably find enacted in real DNNs. This observation is of great practical interest: together with our understanding of noise and nonlinearity it suggests a simple strategy for hyperparameter tuning and model selection which may be a compelling alternative to grid-search approaches; we explore this in Fig. 6 and in further experiments in Supplemental Material (SM) [53].

Feature learning across layers of DNNs—A DNN with L hidden layers, weights $\mathbf{W}_\ell \in \mathbb{R}^{d_\ell \times d_{\ell-1}}$, biases $\mathbf{b}_\ell \in \mathbb{R}^{d_\ell}$, and activation σ maps the input $\mathbf{x}_0 \equiv \mathbf{x}$ to the output (a label) $\mathbf{y} \equiv \mathbf{x}_{L+1}$ via a sequence of intermediate representations $\mathbf{x}_0 \rightarrow \mathbf{x}_1 \rightarrow \dots \rightarrow \mathbf{x}_L \rightarrow \mathbf{x}_{L+1} \equiv \mathbf{y}$, where $\mathbf{x}_{L+1} = F(\mathbf{x}) = \mathbf{W}_{L+1}\mathbf{x}_L + \mathbf{b}_{L+1}$ and

$$\mathbf{h}_\ell = \mathbf{W}_\ell \mathbf{x}_{\ell-1} + \mathbf{b}_\ell, \quad \mathbf{x}_\ell = c_\ell \sigma(\mathbf{h}_\ell), \quad (1)$$

for $\ell = 1, \dots, L$. We call the layers with small ℓ (near the input) *shallow* and those with large L (near the output) *deep*. The activation-dependent normalization factors c_ℓ scale the variance of hidden features in each layer close to 1 [54–56]; they can be replaced by batch normalization [57].

It is natural to expect that in a well-trained DNN the intermediate features $\mathbf{x}_\ell$ improve progressively over ℓ . Following recent work on neural collapse we measure separation as the ratio of variance within and across classes [2–5,41]; in Supplemental Material [53] we show that analogous phenomenology exists in regression. Let $\mathcal{X}_\ell^k$ collect the ℓ th postactivation for examples from class k , with $\bar{\mathbf{x}}_\ell^k$ and N_k its mean and cardinality. The within-class and between-class covariances are

$$\Sigma_\ell^w := \text{Ave}_{n_k}(\text{Cov} \mathcal{X}_\ell^k), \quad \Sigma_\ell^b := \text{Cov}_{n_k}(\bar{\mathbf{x}}_\ell^k)_{k=1}^K, \quad (2)$$

where Ave_{n_k} and Cov_{n_k} are the weighted average and covariance with weight $n_k = N_k/N$. Σ_ℓ^b is the between-class “signal” for classification, and Σ_ℓ^w is the within-class variability. Data separation at layer ℓ is then defined as

$$D_\ell := \log [\text{Tr}(\Sigma_\ell^w)/\text{Tr}(\Sigma_\ell^b)]. \quad (3)$$

The difference $d_\ell = D_{\ell-1} - D_\ell$ represents the contribution of the ℓ th layer. We call the “discrete curve” D_ℓ vs ℓ the *load curve* in anticipation of the mechanical analogy. If indeed each layer improves the data representation the load curve should monotonically decrease.

Modern overparametrized DNNs may perfectly fit the training data for different choices of hyperparameters, while yielding different load curves. One extreme is an RFM where only the last layer learns while the rest are frozen at initialization [9,58]. As the entire load is concentrated on one layer, one might intuitively expect that a more even distribution results in better performance.

A law of data separation?—He and Su [3] show that in many well-trained DNNs the load is distributed uniformly over layers,

$$d_\ell \approx d_{\ell'} \quad \text{for } 1 \leq \ell, \ell' \leq L,$$

giving a linear load curve [59]. This can be proved in linear DNNs with orthogonal initialization and gradient flow training [41,42,60], but as we show below it is brittle: nonlinearity breaks the balance. Equiseparation thus requires additional ingredients; He and Su highlight the importance of an appropriate learning rate.

The noise-nonlinearity phase diagram—We show that DNNs define a family of phase diagrams such that (i) increasing nonlinearity results in increasingly “concave” load curves, with deeper layers learning better features (taking a higher load), $d_\ell < d_{\ell'}$ for $\ell < \ell'$; (ii) noise in the dynamics rebalances the load; and (iii) increasing noise results in convex load curves, with shallower layers learning better features.

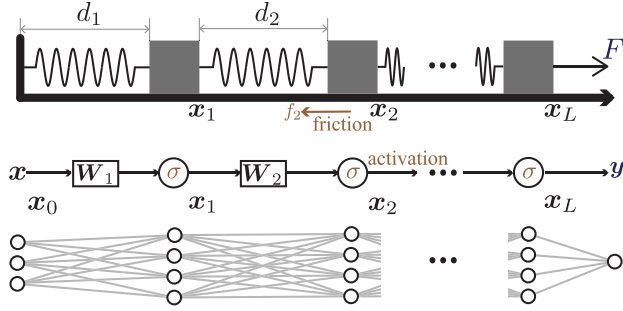


FIG. 2. An illustration of the analogy between a spring-block chain and a deep neural network.

We report these findings in Fig. 1. In all panels the abscissas measure stochasticity and the ordinates non-linearity. We control nonlinearity by varying the slope $\alpha \in [0, 1]$ of the negative part of a leaky rectified linear unit (ReLU) activation, $\text{LReLU}(x) := \max(\alpha x, x)$; $\alpha = 1$ gives a linear DNN, $\alpha = 0$ the ReLU. Consider, for example, Fig. 1(a) where noise is introduced in labels and data (we randomly reset a fraction p of the labels y , and add Gaussian noise with variance p^2 to x_0). Without noise (upper right corner), the load curve is concave; this resembles an RFM or a kernel machine. Increasing noise (right to left) yields a linear and then a convex load curve.

The same phenomenology results from varying learning rate, dropout, and batch size [Figs. 1(b)–1(d)]. Lower learning rate, smaller dropout, and larger batch size all indicate less stochasticity in dynamics. Martin and Mahoney [61] call them temperaturelike parameters; see also Zhou *et al.* [62]. In SM we give simple rules of thumb for trading off these different sources of noise [53]. In all cases high nonlinearity and low noise result in a concave load curve where parameters of deep layers move faster than those of shallow. Low nonlinearity and high noise (bottom left) result in convex load curves. We observe the same behavior for other datasets (e.g., CIFAR10) and network architectures (e.g., CNNs); cf. SM [53]. The goal is to understand how varying these parameter transitions from nonfeature learning models like RFM or NTK to models where all layers learn.

A spring-block theory of feature learning—We now show that the complete phenomenology of feature learning, as seen through layerwise data separation, is mirrored by the stochastic dynamics of a simple spring-block chain. As in Fig. 2, we interpret d_ℓ , the signed elongation of the ℓ th Hookean spring, as data separation by the ℓ th layer. Block movement is impeded by friction which models the effect of dynamical nonlinearity on gradients; noise in the force models stochasticity from minibatching, dropout, or elsewhere.

The equation of motion for the position of the ℓ th block, $x_\ell = \sum_{i=1}^{\ell} d_i$, ignoring block widths, is

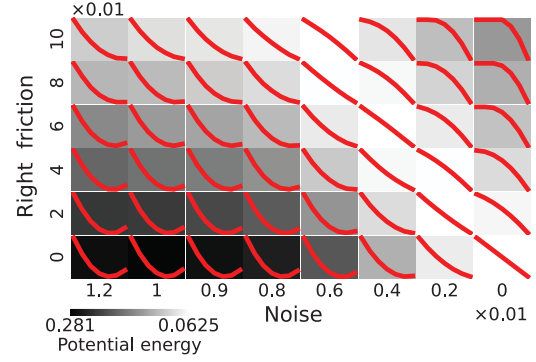


FIG. 3. Phase diagram of the spring-block system (5) for friction μ_+ vs noise level ϵ . We set $k = 1$, $\mu_- = 0.2$, and $L = 7$. The load curves ($D_\ell = y - x_\ell$) are recorded at $t = 100$; the shading corresponds to the elastic potential energy.

$$\ddot{x}_\ell = k(x_{\ell+1} - 2x_\ell + x_{\ell-1}) - \gamma \dot{x}_\ell + f_\ell + \epsilon \xi_\ell, \quad (4)$$

for $\ell = 1, \dots, L$, where we assumed unit masses, k is the elastic modulus, γ is a linear damping coefficient sufficiently large to avoid oscillations, ϵ controls the noise strength, and ξ_ℓ is noise such that $\langle \xi_\ell(t), \xi_\ell(s) \rangle = \delta(t - s)$. As DNNs at initialization do not separate data we let $x_\ell(0) = 0$. The dynamics is driven by force applied to the last block $F = k(y - x_L)$. We set $x_0 \equiv 0$ and $x_{L+1} \equiv y$ to model training data and targets. The load curve plots the distance of the ℓ th block from the target $D_\ell = y - x_\ell$. It reflects how much the ℓ th spring—or the ℓ th layer—contributes to “explaining” the total extension—or the total data separation. One key insight is that the friction must be asymmetric to model the propagation of noise during training as we elaborate below. We set the sliding and maximum static friction to μ_+ for rightward and μ_- for leftward movement. In this model the friction acts on noisy force. If noise is added to the velocity independently of the friction we obtain a more standard (but physically less realistic) Langevin dynamics which exhibits similar qualitative behavior; for additional details see SM [53].

The spring-block dynamics of data separation in DNNs—We now show experimentally and analytically that the proposed model results in a phenomenology analogous to that of data separation in real networks. There is a striking similarity between the phase diagram of the spring-block model in Fig. 3 and the DNN phase diagrams in Fig. 1. Not only are the equilibria of the two systems similar, but also the stochastic dynamics; we show this in Fig. 4.

Nonlinearity breaks the separation balance: We first show how our model explains concave load curves. For simplicity we work in the overdamped approximation $\gamma \gg 1$; in SM [53] we show that the second order system has the same qualitative behavior. Scaling time by γ , Eq. (4) yields

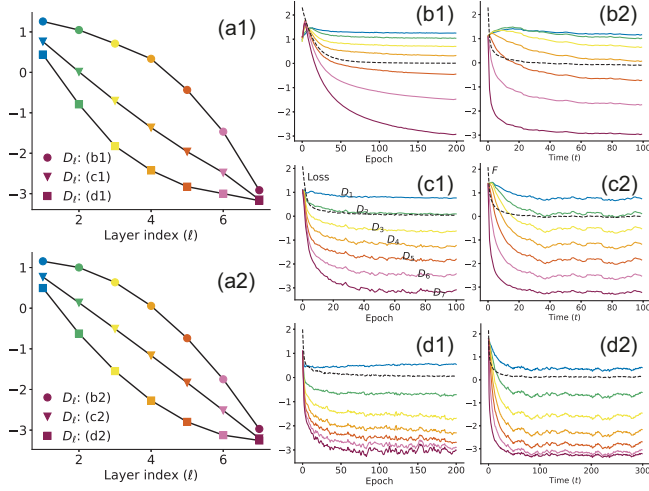


FIG. 4. The load curves at convergence (a) and trajectories (b)–(d) for a seven-hidden-layer ReLU MLP on MNIST (1) vs our spring-block model (2). For the MLP (1), the ordinate is D_ℓ (data separation at layer ℓ); the dashed line is the training loss. Characteristic behaviors: (b) high nonlinearity (high friction) and low randomness in training (noise in force); (c) balanced nonlinearity and randomness; (d) low nonlinearity and strong randomness. In the spring system (2), the ordinate is the distance to the target $D_\ell = y - x_\ell$. The values are scaled to match the same regime used in the DNN for ease of visualization. The dashed line is the force at the rightmost block F which is at a different scale from D_ℓ .

$$\dot{x}_\ell = \sigma(k(Lx)_\ell + \varepsilon \xi_\ell) \quad (5)$$

where $(Lx)_\ell := x_{\ell+1} - 2x_\ell + x_{\ell-1}$ and $\sigma(z) = 0$ if $-\mu_- \leq z \leq \mu_+$; $z - \mu_+$ if $z > \mu_+$; and $z + \mu_-$ if $z < -\mu_-$.

Without noise and friction ($\varepsilon = 0$, $\mu_- = \mu_+ = 0$) the system is linear with the trivial unique equilibrium $d_\ell^* \equiv y/(L+1)$ for all ℓ , which corresponds to the state of lowest elastic potential energy. However, analyzing the resulting system of ODEs shows that adding friction in (5) immediately breaks the symmetry and results in an *unbalanced* equilibrium,

$$x_\ell^* = \frac{y\ell}{L+1} - \frac{\mu_+}{k} \left(\frac{L\ell}{2} - \frac{\ell(\ell-1)}{2} \right) \quad (6)$$

if we assume the initial elastic potential $k(y - x_0)^2/2$ is sufficiently large such that all blocks eventually move. In this case $\Delta d_\ell^* := d_{\ell+1}^* - d_\ell^* = (Lx^*)_\ell = \mu_+/k > 0$ and the load curve is concave. Note that this result only involves μ_+ but not μ_- as without noise and with sufficient damping the blocks only move to the right. The interpretation is that in equilibrium, the friction at each block absorbs some of the load so that the shallower springs extend less; in a DNN this corresponds to a regime where the deepest layer, whose gradients do not experience nonlinearity, does most of the separation.

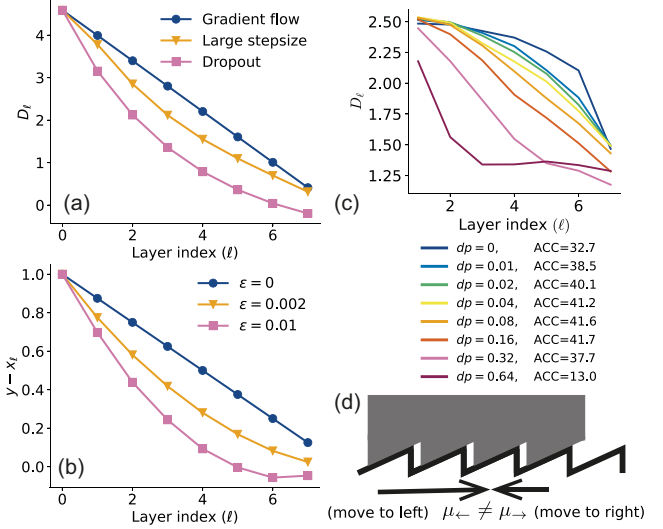


FIG. 5. (a) Load curve of a deep linear neural network on random orthonormal datasets. (b) Spring-block model without left friction ($\mu_+ = 0$) with $\mu_- = 0.1$, under varying levels of noise ε . (c) Load curve of seven-layer CNN trained on the CIFAR10 dataset; db and ACC denote the dropout ratio and test accuracy. (d) An illustration of asymmetric friction.

Noise reequilibrates the load: If friction reduces load, how can a chain with friction—or a nonlinear DNN—result in a uniformly distributed load? We know from Fig. 1 that in DNNs stochastic training helps achieve this. We now show how our model reproduces this behavior and in particular a counterintuitive phenomenon in Fig. 1 where large noise results in *convex* load curves where shallow layers learn better than deep. We show that this happens when $\mu_- > \mu_+$.

We begin by defining the *effective* friction over a time window of length η . Let $\bar{\varepsilon} := \varepsilon/\sqrt{\eta}$ and $\zeta_t \sim (\xi_{t+\eta} - \xi_t)/\sqrt{\eta}$ (iid). We will assume for convenience that the increments ζ are bounded (see SM [53] for details). The effective friction is then

$$\mu_{\text{eff}} := \mathbb{E}_\zeta \{ [\sigma(k\Delta d + \bar{\varepsilon}\zeta) - k\Delta d] \sigma(k\Delta d + \bar{\varepsilon}\zeta) \neq 0 \}.$$

For large noise we will have that $\mu_{\text{eff}} \approx \lim_{\bar{\varepsilon} \rightarrow \infty} \mu_{\text{eff}} = \frac{1}{2}(\mu_+ - \mu_-)$. Since this effective friction is approximately independent of x , the load curve can be approximated by substituting μ_{eff} for μ_+ in Eq. (6), which leads to

$$\Delta d^* \approx \frac{1}{k} \lim_{\bar{\varepsilon} \rightarrow \infty} \mu_{\text{eff}}(\bar{\varepsilon}) = \frac{\mu_+ - \mu_-}{2k}. \quad (7)$$

It is now clear that with sufficient noise the load curve is concave if $\mu_+ > \mu_-$, linear if $\mu_+ = \mu_-$, and convex if $\mu_+ < \mu_-$. We can also see that for $\bar{\varepsilon} = 0$, $\mu_{\text{eff}} = \mu_+$, so that this effective friction correctly generalizes the noiseless case Eq. (6) and we have that $\Delta d^* \in \{[(\mu_+ - \mu_-)/2k], (\mu_+/k)\}$.

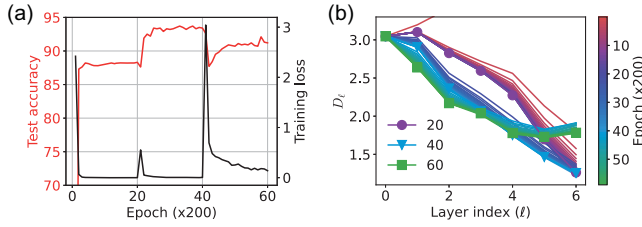


FIG. 6. The dynamics of load curves for a deep CNN. (a) Test accuracy versus training loss. (b) The corresponding load curves during training. In the experiments, we introduce 5% at epoch 20×200 and 30% dropout at epoch 40×200 .

Further, when $\Delta d \in \{[(\mu_{\rightarrow} - \mu_{\leftarrow})/2k], (\mu_{\rightarrow}/k)\}$, it holds that $(d\mu_{\text{eff}}/d\varepsilon) \leq 0$. It implies that increasing noise always decreases effective friction. This resembles phenomena like acoustic lubrication or noise-induced superlubricity [63]. When $\mu_{\leftarrow} > \mu_{\rightarrow} > 0$, as we vary ε from 0 to ∞ we will first see a concave, then a linear, and finally a convex load curve. Therefore, when $\mu_{\leftarrow} > \mu_{\rightarrow} > 0$ our model explains the entire DNN phase diagram.

How can we relate the condition $\mu_{\leftarrow} > \mu_{\rightarrow} > 0$ to DNN phenomenology? Note that in our model $\mu_{\leftarrow}$ is activated only due to the noise, since the signal (the pulling force) is always to the right. In a DNN, the forward pass of the backpropagation algorithm computes the activations while the backward pass computes the gradients. In the forward pass, the input is multiplied by the weight matrices starting from the shallowest to the deepest. With noise (e.g., dropout), the activations of the deepest layer accumulate the largest noise. Thus, noise in DNN training chiefly propagates from shallow to deep, yielding noisiest gradients in deepest layers. This is exactly what $\mu_{\leftarrow} > \mu_{\rightarrow}$ in our model implies, since $\mu_{\leftarrow}$ is only triggered by noise.

Indeed, from Eq. (7) we see that when only friction for leftward movement is present, $\mu_{\leftarrow} > \mu_{\rightarrow} = 0$, the load curve is convex if $\varepsilon > 0$ and linear if $\varepsilon = 0$ [Fig. 5(b)]. The equilibrium is independent of $\mu_{\leftarrow}$ since no block moves left. This parallels findings in linear DNNs where training with gradient flow and “whitened” data lead to a linear load curve [41]. In contrast, introducing noise makes the load curve convex [Fig. 5(a)]. This shows that *dynamical* nonlinearity—or friction in our model—exists even in linear DNNs, so that their learning dynamics are nonlinear [36].

Equiseparation minimizes elastic potential energy and improves generalization: We finally show how our theory gives insight into generalization. Among all spring-block chains under a fixed load, the equiseparated one is the most stable in the sense of having the lowest potential energy. This motivates us to study the test accuracy of DNNs—shown as background shading in Fig. 1—as a function of load curve curvature. The result is intriguing: linear load curves correspond to the highest test accuracy. It suggests that by balancing nonlinearity with noise, DNNs are at once highly expressive and not overfitting. That a uniform load

distribution yields the best performance is intuitively pleasing, but it is also valuable operationally as it may help guide training to find better networks. In Fig. 6 we show a proof-of-concept example with a CNN: initial training yields a concave load curve. Our theory suggests that more noise in training would flatten the load curve and improve generalization—which is indeed what happens. Further details and experiments with very deep networks can be found in SM [53].

Conclusion—Deep learning theories are mostly built bottom up, but fields like physics, biology, neuroscience, and economics benefit from both bottom-up and top-down, phenomenological approaches. Mechanical analogies such as spring-block models play an important role across science; a prime example is the Burridge-Knopoff model in seismology [64] and related ideas in neuroscience [65]. We think that deep learning can similarly benefit from both paradigms, especially with complex phenomena like feature learning which require to simultaneously consider depth, nonlinearity, noise and data.

Our phenomenological model elucidates the role of nonlinearity and randomness and suggests exciting connections with generalization that may inspire new theory, but also, importantly, new approaches to hyperparameter tuning and model selection. The current theory applies to DNNs where inner layers are alike; extensions to heterogeneous DNNs will require refined definitions of data separation and new mechanical models.

We finally mention that we considered various cascade structures as possible analogies to DNNs; the SM links to real experiments with a folding ruler [53].

Acknowledgments—We are grateful to Hangfeng He, Weijie Su, and Qing Qu for inspiring discussions about data separation. C.S. and I.D. were supported by the ERC Starting Grant No. 852821—SWING. L.P. acknowledges support from National Natural Science Foundation of China (NSFC), Grants No. 62006122 and No. 42230406.

- [1] M. D. Zeiler and R. Fergus, Visualizing and understanding convolutional networks, in *Proceedings of the Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland 2014, Part I 13* (Springer, New York, 2014), pp. 818–833.
- [2] V. Pappas, X. Han, and D. L. Donoho, Prevalence of neural collapse during the terminal phase of deep learning training, *Proc. Natl. Acad. Sci. U.S.A.* **117**, 24652 (2020).
- [3] H. He and W. J. Su, A law of data separation in deep learning, *Proc. Natl. Acad. Sci. U.S.A.* **120**, e2221704120 (2023).
- [4] J. Zarka, F. Guth, and S. Mallat, Separation and concentration in deep networks, in *Proceedings of the ICLR 2021-9th International Conference on Learning Representations* (OpenReview.net, Vienna, Austria, 2021).
- [5] A. Rangamani, M. Lindegaard, T. Galanti, and T. A. Poggio, Feature learning in deep classifiers through intermediate

- neural collapse, in *Proceedings of the International Conference on Machine Learning* (PMLR, Honolulu, 2023), pp. 28729–28745.
- [6] V. Kothapalli, T. Tirer, and J. Bruna, A neural collapse perspective on feature evolution in graph neural networks, *Adv. Neural Inf. Process. Syst.* **36**, 14134 (2023).
 - [7] S. Qin, N. Mudur, and C. Pehlevan, Contrastive similarity matching for supervised learning, *Neural Comput.* **33**, 1300 (2021).
 - [8] T. Hofmann, B. Schölkopf, and A. J. Smola, Kernel methods in machine learning, *Ann. Stat.* **36**, 1171 (2008).
 - [9] A. Rahimi and B. Recht, Random features for large-scale kernel machines, *Adv. Neural Inf. Process. Syst.* **20**, 1177 (2007).
 - [10] A. Daniely, R. Frostig, and Y. Singer, Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity, *Adv. Neural Inf. Process. Syst.* **29**, 2261 (2016).
 - [11] A. Radhakrishnan, D. Beaglehole, P. Pandit, and M. Belkin, Mechanism for feature learning in neural networks and backpropagation-free machine learning models, *Science* **383**, 1461 (2024).
 - [12] Y. Lou, C. E. Mingard, and S. Hayou, Feature learning and signal propagation in deep neural networks, in *Proceedings of the International Conference on Machine Learning* (PMLR, Baltimore, 2022), pp. 14248–14282.
 - [13] A. Baratin, T. George, C. Laurent, R. D. Hjelm, G. Lajoie, P. Vincent, and S. Lacoste-Julien, Implicit regularization via neural feature alignment, in *Proceedings of the International Conference on Artificial Intelligence and Statistics* (PMLR, 2021), pp. 2269–2277.
 - [14] G. Yang and E. J. Hu, Tensor programs IV: Feature learning in infinite-width neural networks, in *Proceedings of the International Conference on Machine Learning* (PMLR, 2021), pp. 11727–11737.
 - [15] L. Chizat and P. Netrapalli, Steering deep feature learning with backward aligned feature updates, [arXiv:2311.18718](https://arxiv.org/abs/2311.18718).
 - [16] S. Mei, A. Montanari, and P.-M. Nguyen, A mean field view of the landscape of two-layer neural networks, *Proc. Natl. Acad. Sci. U.S.A.* **115**, E7665 (2018).
 - [17] L. Chizat and F. Bach, On the global convergence of gradient descent for over-parametrized models using optimal transport, *Adv. Neural Inf. Process. Syst.* **31**, 3040 (2018).
 - [18] Y. Wang, J. Lacotte, and M. Pilanci, The hidden convex optimization landscape of regularized two-layer relu networks: An exact characterization of optimal solutions, in *Proceedings of the International Conference on Learning Representations* (OpenReview.net, Vienna, Austria, 2021).
 - [19] L. Arnaboldi, L. Stephan, F. Krzakala, and B. Loureiro, From high-dimensional & mean-field dynamics to dimensionless odes: A unifying approach to sgd in two-layers networks, in *Proceedings of the Thirty Sixth Annual Conference on Learning Theory* (PMLR, Bangalore, 2023), pp. 1199–1227.
 - [20] T. Luo, Z.-Q. J. Xu, Z. Ma, and Y. Zhang, Phase diagram for two-layer relu neural networks at infinite-width limit, *J. Mach. Learn. Res.* **22**, 1 (2021).
 - [21] A. Maillard, A. S. Bandeira, D. Belius, I. Dokmanić, and S. Nakajima, Injectivity of relu networks: Perspectives from statistical physics, [arXiv:2302.14112](https://arxiv.org/abs/2302.14112).
 - [22] R. Pacelli, S. Ariosto, M. Pastore, F. Ginelli, M. Gherardi, and P. Rotondo, A statistical mechanics framework for bayesian deep neural networks beyond the infinite-width limit, *Nat. Mach. Intell.* **5**, 1497 (2023).
 - [23] P. Baglioni, R. Pacelli, R. Aiudi, F. Di Renzo, A. Vezzani, R. Burioni, and P. Rotondo, Predictive power of a bayesian effective action for fully connected one hidden layer neural networks in the proportional limit, *Phys. Rev. Lett.* **133**, 027301 (2024).
 - [24] H. Cui, L. Pesce, Y. Dandi, F. Krzakala, Y. M. Lu, L. Zdeborová, and B. Loureiro, Asymptotics of feature learning in two-layer networks after one gradient-step, *arXiv preprint arXiv:2402.04980* (2024).
 - [25] J. Sohl-Dickstein, The boundary of neural network trainability is fractal, *arXiv preprint arXiv:2402.06184* (2024).
 - [26] R. Marino and F. Ricci-Tersenghi, Phase transitions in the mini-batch size for sparse and dense two-layer neural networks, *Mach. Learn.* **5**, 015015 (2024).
 - [27] Y. Dandi, E. Troiani, L. Arnaboldi, L. Pesce, L. Zdeborová, and F. Krzakala, The benefits of reusing batches for gradient descent in two-layer networks: Breaking the curse of information and leap exponents, *arXiv preprint arXiv:2402.03220* (2024).
 - [28] S. Ciceri, L. Cassani, M. Osella, P. Rotondo, F. Valle, and M. Gherardi, Inversion dynamics of class manifolds in deep learning reveals tradeoffs underlying generalization, *Nat. Mach. Intell.* **6**, 40 (2024).
 - [29] E. Boursier and N. Flammarion, Early alignment in two-layer networks training is a two-edged sword, [arXiv:2401.10791](https://arxiv.org/abs/2401.10791).
 - [30] S. Goldt, M. Mézard, F. Krzakala, and L. Zdeborová, Modeling the influence of data structure on learning in neural networks: The hidden manifold model, *Phys. Rev. X* **10**, 041044 (2020).
 - [31] A. Jacot, F. Gabriel, and C. Hongler, Neural tangent kernel: Convergence and generalization in neural networks, *Adv. Neural Inf. Process. Syst.* **31**, 8580 (2018).
 - [32] F. Guth, B. Ménard, G. Rochette, and S. Mallat, A rainbow in deep network black boxes, [arXiv:2305.18512](https://arxiv.org/abs/2305.18512).
 - [33] K. H. R. Chan, Y. Yu, C. You, H. Qi, J. Wright, and Y. Ma, ReduNet: A white-box deep network from the principle of maximizing rate reduction, *J. Mach. Learn. Res.* **23**, 1 (2022).
 - [34] C. Fang, H. He, Q. Long, and W. J. Su, Exploring deep neural networks via layer-peeled model: Minority collapse in imbalanced training, *Proc. Natl. Acad. Sci. U.S.A.* **118**, e2103091118 (2021).
 - [35] S. Arora, N. Cohen, N. Golowich, and W. Hu, A convergence analysis of gradient descent for deep linear neural networks, [arXiv:1810.02281](https://arxiv.org/abs/1810.02281).
 - [36] Q. Li and H. Sompolinsky, Statistical mechanics of deep linear neural networks: The backpropagating kernel renormalization, *Phys. Rev. X* **11**, 031059 (2021).
 - [37] B. Bordonon and C. Pehlevan, Self-consistent dynamical field theory of kernel evolution in wide neural networks, *Adv. Neural Inf. Process. Syst.* **35**, 32240 (2022).
 - [38] B. Bordonon, A. Atanasov, and C. Pehlevan, A dynamical model of neural scaling laws, [arXiv:2402.01092](https://arxiv.org/abs/2402.01092).
 - [39] G. Ben Arous, R. Gheissari, and A. Jagannath, High-dimensional limit theorems for SGD: Effective dynamics

- and critical scaling, *Adv. Neural Inf. Process. Syst.* **35**, 25349 (2022).
- [40] G. B. Arous, R. Gheissari, J. Huang, and A. Jagannath, High-dimensional SGD aligns with emerging outlier eigenspaces, [arXiv:2310.03010](#).
- [41] C. Yaras, P. Wang, W. Hu, Z. Zhu, L. Balzano, and Q. Qu, The law of parsimony in gradient descent for learning deep linear networks, [arXiv:2306.01154](#).
- [42] P. Wang, X. Li, C. Yaras, Z. Zhu, L. Balzano, W. Hu, and Q. Qu, Understanding deep representation learning via layer-wise feature compression and discrimination, [arXiv:2311.02960](#).
- [43] E. Gardner and B. Derrida, Optimal storage properties of neural network models, *J. Phys. A* **21**, 271 (1988).
- [44] C. Baldassi, C. Lauditi, E. M. Malatesta, G. Perugini, and R. Zecchina, Unveiling the structure of wide flat minima in neural networks, *Phys. Rev. Lett.* **127**, 278301 (2021).
- [45] S. Becker, Y. Zhang, and A. A. Lee, Geometry of energy landscapes and the optimizability of deep neural networks, *Phys. Rev. Lett.* **124**, 108301 (2020).
- [46] C. Shi, L. Pan, H. Hu, and I. Dokmanić, Homophily modulates double descent generalization in graph convolution networks, *Proc. Natl. Acad. Sci. U.S.A.* **121**, e2309504121 (2024).
- [47] J. Pennington and Y. Bahri, Geometry of neural network loss surfaces via random matrix theory, in *Proceedings of the International conference on machine learning* (PMLR, Sydney, 2017), pp. 2798–2806.
- [48] X. Fernández-Real and A. Figalli, The continuous formulation of shallow neural networks as wasserstein-type gradient flows, in *Proceedings of the Analysis at Large: Dedicated to the Life and Work of Jean Bourgain* (Springer, New York, 2022), pp. 29–57.
- [49] N. Yang, C. Tang, and Y. Tu, Stochastic gradient descent introduces an effective landscape-dependent regularization favoring flat solutions, *Phys. Rev. Lett.* **130**, 237101 (2023).
- [50] A. Sclocchi and M. Wyart, On the different regimes of stochastic gradient descent, *Proc. Natl. Acad. Sci. U.S.A.* **121**, e2316301121 (2024).
- [51] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, *J. Mach. Learn. Res.* **15**, 1929 (2014).
- [52] S. Sukhbaatar and R. Fergus, Learning from noisy labels with deep neural networks, [arXiv:1406.2080](#).
- [53] See Supplemental Material at <http://link.aps.org/supplemental/10.1103/ys4n-2tj3>, which includes: (1) a folding ruler experiment, (2) additional experimental results on various networks and datasets, (3) details of numerical experiments and reproducibility, (4) some other metrics for data separation of intermediate features, (5) regression, (6) additional details about the spring block systems and (7) scaling and quantity of noise sources.
- [54] K. He, X. Zhang, S. Ren, and J. Sun, Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in *Proceedings of the IEEE international conference on computer vision* (IEEE Computer Society, Santiago, Chile, 2015), pp. 1026–1034.
- [55] B. Poole, S. Lahiri, M. Raghu, J. Sohl-Dickstein, and S. Ganguli, Exponential expressivity in deep neural networks through transient chaos, *Adv. Neural Inf. Process. Syst.* **29**, 3368 (2016).
- [56] S. S. Schoenholz, J. Gilmer, S. Ganguli, and J. Sohl-Dickstein, Deep information propagation, [arXiv:1611.01232](#).
- [57] S. Ioffe and C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in *Proceedings of the International Conference on Machine Learning* (PMLR, Lille, 448–456).
- [58] H. Hu and Y. M. Lu, Universality laws for high-dimensional learning with random features, *IEEE Trans. Theory* **69**, 1932 (2022).
- [59] He and Su [3] and some other previous works [2,4] define the data separation slightly differently as $\hat{D}_\ell := \log [\text{Tr}(\Sigma_\ell^w \Sigma_\ell^{b+})]$, where Σ^+ denotes the Moore-Penrose inverse of Σ . As mentioned in [42], Eq. (3) can be regarded as a simplified version of the above definition.
- [60] S. Arora, N. Cohen, and E. Hazan, On the optimization of deep networks: Implicit acceleration by overparametrization, in *International conference on machine learning* (PMLR, Stockholm, 2018), pp. 244–253.
- [61] C. H. Martin and M. W. Mahoney, Rethinking generalization requires revisiting old ideas: Statistical mechanics approaches and complex learning behavior, [arXiv:1710.09553](#).
- [62] Y. Zhou, T. Pang, K. Liu, M. W. Mahoney, Y. Yang *et al.*, Temperature balancing, layer-wise weight analysis, and neural network training, *Adv. Neural Inf. Process. Syst.* **36**, 63542 (2024).
- [63] S. Shi, D. Guo, and J. Luo, Micro/atomic-scale vibration induced superlubricity, *Frict. Wear Mach.* **9**, 1163 (2021).
- [64] R. Burridge and L. Knopoff, Model and theoretical seismicity, *Bull. Seismol. Soc. Am.* **57**, 341 (1967).
- [65] J. J. Hopfield, Neurons, dynamics and computation, *Phys. Today* **47**, No. 2, 40 (1994).