

# Neural networks become centroid classifiers when trained on heavily corrupted inputs

Justin Tahmassebpur<sup>1,\*</sup>, Asadullah Bhuiyan<sup>2</sup>, Hyejin Kim<sup>2</sup>, and Omri Lesser<sup>2</sup>

<sup>1</sup>*Department of Applied and Engineering Physics, Cornell University, Ithaca, New York 14853, USA*

<sup>2</sup>*Department of Physics, Cornell University, Ithaca, New York 14853, USA*



(Received 15 June 2026; accepted 18 September 2026; published 23 October 2026)

Learning from imperfect data is a central theme in machine learning, connecting practical questions of robustness to fundamental questions of learnability. Here, we examine attribute noise: learning from corrupted inputs while keeping the labels intact, a setting that has received considerably less analytical attention than its label-noise counterpart. We consider two types of corruption models: additive noise and replacement noise. Through experiments with multilayer perceptrons (MLPs) on corrupted classification datasets, we find that neural networks remain robust, maintaining well-above-chance accuracy even when inputs are >90% corrupted—far beyond human recognition. To understand this robustness, we analyze infinite-width networks in the heavy-corruption regime using a mean-field-inspired approach and derive a leading-order decision rule for the classification outcome: The network implements a prototype rule, the nearest-class-mean, assigning each test point to the class whose training-set average it most closely resembles. This leading-order decision rule is universal across a broad range of MLP architectures, holding for any depth, as well as a wide class of activation functions and noise distributions. The same centroid mechanism closely matches finite-width network behavior in our experiments and provides an interpretable and analytically tractable account of why learning can succeed even when individual training examples carry almost no signal.

DOI: [10.1103/qkrg-wbbj](https://doi.org/10.1103/qkrg-wbbj)

## I. INTRODUCTION

Machine learning is fundamentally reshaping science and technology, raising profound questions about both its capabilities and its limitations. A central challenge in this context is learning from imperfect data: How can a model generalize when the examples available during training are degraded? The canonical setting is that of *label noise*, where the target labels in a classification or regression task are corrupted [1–7]. The complementary problem of *attribute noise*—where the labels remain correct but the input features themselves are corrupted—has received considerably less analytical attention. Furthermore, attribute noise is common in practice: Measured attributes are often degraded by sensor noise, acquisition errors, missing features, or other imperfections, while the associated labels may remain reliable [8].

Attribute noise has been examined from several perspectives. Early work showed that injecting noise into inputs during training can act as a regularizer and improve generalization [9–16]. Later work addressed feature corruption and deletion more directly, developing classifiers that are robust to missing inputs and training procedures that explicitly marginalize over corrupted features [17–19]. A separate line of research considers *test-time* corruption, asking how models

trained on clean data degrade when evaluated on corrupted inputs [20]. This form of robustness is quantified by benchmarks such as ImageNet-C [21] and MNIST-C [22]. Together, these bodies of work address several important aspects of the robustness of corruption. However, they leave open a basic question: Can neural networks trained on *severely* corrupted attributes, but clean labels, learn a coherent classification rule and generalize well to clean test data?

In this paper, we show that the answer is surprisingly “yes.” Empirically, we train multilayer perceptrons (MLPs) on corrupted versions of the standard image datasets MNIST [23], KMNIST [24], and Fashion-MNIST [25], using two corruption models appropriate for image data: pixel replacement noise and additive noise. Across all three datasets, we find that the networks remain strikingly robust even when the training inputs are severely corrupted (see Fig. 1). At such high-noise levels, the images are nearly indistinguishable to the human eye, yet the network can still learn to classify clean test images far better than chance.

We then explain this robustness analytically using the neural tangent kernel (NTK) framework [26]. For any wide MLP trained on arbitrary datasets with clean labels and generic independent and identically distributed (i.i.d.) input corruption, we derive an explicit analytical form for the learned predictor in the high-noise limit. The resulting high-noise decision rule is simple: The analytical form of the predictor is a linear nearest-class-mean—or *centroid* [27]—classifier on top of a competing stochastic background.

This theoretical result is then compared to numerical experiments on finite-width MLPs, and we find excellent agreement for both raw logit outputs and the argmax-filtered test accuracy. This empirical agreement with finite-width networks

\*Contact author: jt577@cornell.edu

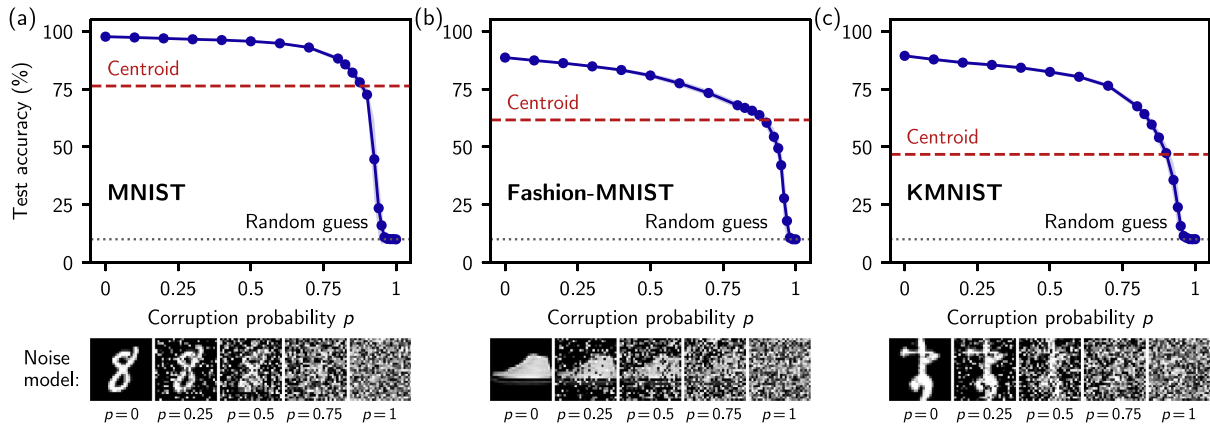


FIG. 1. Test accuracy vs corruption probability  $p$  for width-128, three-layer ReLU MLPs trained with cross-entropy loss on images with i.i.d. pixel replacement noise (blue solid lines). Red dashed lines show dataset-averaged test accuracy computed from the centroid classifier [Eq. (15)] on the clean data—we provide this as a baseline. Three class-balanced datasets are shown: (a) MNIST, (b) Fashion-MNIST, and (c) KMNIST. Below each panel, we demonstrate the noise model by showing representative examples of images subject to replacement noise at increasing  $p$ . For each  $p$ , an MLP is trained on a dataset where each pixel is independently replaced with uniform noise with probability  $p$ ; the test set is left clean. Accuracy metrics are averaged over 100 independent noisy training realizations per  $p$  value, and shaded bands surrounding each curve represent one standard deviation across realizations (see Appendix I for a discussion of the nonmonotonic behavior of the standard deviation). Even when the replacement noise has corrupted up to 80% ( $p = 0.8$ ) of each image in the training set, test accuracy remains at or above  $\sim 70\%$  across all three datasets, with accuracy collapsing toward uniform random guessing ( $100/C = 10\%$  for  $C = 10$  classes, dashed line) only near  $p \rightarrow 1$ . Furthermore, the centroid baseline agrees with the noisy test accuracy near  $p \sim 0.9$ .

suggests that their surprising robustness on the image datasets considered here comes not from a complicated denoising or feature-recovery procedure, but from aggregating the weak class information that survives the corruption into a prototype-based decision rule—the centroid rule.

Furthermore, the leading form of the high-noise decision rule is independent of depth, activation, and the details of the chosen noise distribution. The origin of this universality can be understood through the lens of symmetry and linear response: At pure noise, training examples become exchangeable, and the leading class-dependent response to a small surviving nonexchangeable signal is forced to depend on centered-class averages.

These results connect our work to prototype-based views of classification, which have appeared in both classical and modern settings, including nearest-class-mean methods, prototypical networks, and recent analyses of class-mean structure in deep representations [27–32]. Our setting differs in emphasis: We do not assume a prototype-based model from the outset, nor do we study prototype geometry as an empirical tendency of learned representations. Instead, we *derive* the prototype rule analytically as the effective predictor of a wide network trained in the high-attribute-noise regime. More broadly, our results identify a rare regime in which neural networks can be understood in unusually explicit terms: Despite the complexity of the architecture, the learned rule itself becomes simple, universal, and interpretable.

The remainder of this paper is organized as follows. In Sec. II, we introduce the two corruption models and present our motivating empirical finding: MLPs classify clean test data well above chance even with heavy train-time corruption. In addition, we present results on MLP classification with independent test-time *and* train-time corruption, finding that classification power is optimized when the training-set corruption strength matches that of the test set. Following

this, we derive a theoretical prediction for the trained network output in the high-noise regime using the NTK framework and compare with numerical experiments on finite-width MLPs in Sec. III. Furthermore, we present a symmetry argument for the origin of the decision rule and its universality across noisy MLP architectures. Finally, Sec. IV discusses the implications of our results and outlook. Details on numerical experiments, technical derivations, and supporting details are collected in the Appendixes.

## II. FEATURE CORRUPTION MODELS AND EMPIRICAL RESULTS

In this section, we present an empirical study of multi-class classification with MLPs trained on feature-corrupted inputs. We will first present the corruption channels of interest, followed by a presentation of empirical results for MLP performance when trained on correctly labeled but feature-corrupted inputs, and tested on clean data; this setup will be treated analytically in Sec. III. In addition, we briefly present empirical results for MLP performance when both the training *and* testing examples are independently corrupted, with further details provided in Appendix B.

Empirical findings presented in this section were obtained from fully connected neural networks with three hidden layers of width 128, rectified linear unit (ReLU) activation, trained for 20 epochs using the Adam optimizer with batch size 128 and cross-entropy loss, and averaged over 100 random seeds. Further details on the experiments can be found in Table I in Appendix A.

### A. Corruption models and setup

We begin by introducing the notation and noise models used throughout this work. Let  $x_{k\chi}$  denote the  $k$ th feature

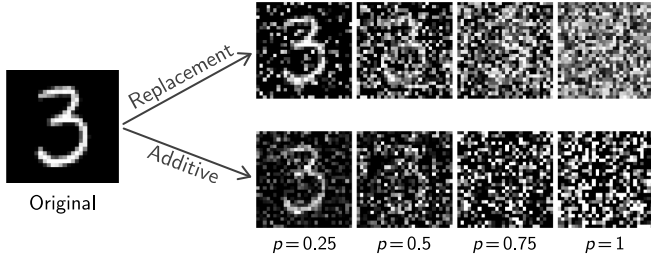


FIG. 2. Visualization of the two corruption models used in this work applied to an MNIST image. The top panel shows the replacement-noise model, where each pixel is independently replaced with uniform noise with probability  $p$  [see Eq. (1)]. The bottom panel shows the additive noise model, where each pixel is replaced by a convex combination of the clean pixel and an independent noise variable, with mixing parameter  $p$  [see Eq. (2)]. The panels show increasing corruption levels  $p$ .

of *clean* training example  $\chi$ , where  $k = 1, \dots, d$  indexes the feature dimension and  $\chi = 1, \dots, N$  indexes the training set. The corresponding corrupted data point is denoted by  $\tilde{x}_{k\chi}$ . Our main theoretical analysis focuses on the large-feature-dimension regime  $d \gg 1$ , which is natural for image data and many modern high-dimensional learning settings.

We consider two corruption models, as shown in Fig. 2. The first is *replacement noise*,

$$\tilde{x}_{k\chi} = (1 - b_{k\chi})x_{k\chi} + b_{k\chi}u_{k\chi}, \quad (1)$$

where  $b_{k\chi}$  are i.i.d. Bernoulli random variables with  $\Pr(b_{k\chi} = 1) = p$ , so that each feature (each pixel in image data) is independently replaced with probability  $p$ , and  $u_{k\chi}$  is an i.i.d. noise variable. In our numerical experiments, we take  $u_{k\chi} \sim \text{Unif}[0, 1]$ , which is a natural choice for grayscale image data such as MNIST, where each of the  $28 \times 28$  pixels takes on a value in  $[0, 1]$ . The second corruption model is *additive noise*,

$$\tilde{x}_{k\chi} = (1 - p)x_{k\chi} + p\xi_{k\chi}, \quad (2)$$

where  $\xi_{k\chi}$  is an i.i.d. noise variable. Notice that in both models,  $p = 0$  corresponds to the clean case and  $p = 1$  is the maximally corrupted case, where the training data become completely random. The empirical results in the main text focus on replacement noise, while additive-Gaussian noise is treated in Appendix E.

Although our empirical results use specific distributions (uniform and Gaussian), we show in Appendix C that the analytical framework applies more broadly to generic  $u_{k\chi}$ ,  $\xi_{k\chi}$  drawn i.i.d. from *any* distribution, since only the first two moments of the distribution enter the calculation of the signal. Equivalently, a specific choice of noise distribution can be replaced by a Gaussian with matched mean and variance without altering the result. This is similar in spirit to the Gaussian equivalence principle from the high-dimensional learning literature [33–36], under which the training and generalization behavior of a high-dimensional model depends on the data distribution only through its first two moments.

Importantly, the labels are assumed to remain clean throughout training. We will denote the label vector associated with class  $i = 1, \dots, C$  as  $y_{i\chi}$ , with  $y_{i\chi}$  one-hot in the class index. We assume that the dataset is class balanced, so that each

of the  $C$  classes contains the same number of training examples. This assumption is not essential; it is adopted here only to streamline the discussion and is relaxed in Appendix C.

## B. Robustness to training-set corruption

Figure 1 summarizes the basic empirical phenomenon. Using the established image datasets MNIST, Fashion-MNIST, and KMNIST [23–25], we show the test accuracy of trained MLPs as a function of the corruption probability  $p$  for replacement-noise corrupted training data, tested on *clean* data. Representative corrupted images are displayed below each panel. By  $p = 0.75$ , the training images are already extremely difficult for a human to recognize, yet the networks still achieve strikingly high clean test accuracy: around 80%–90% on MNIST and 70%–80% on Fashion-MNIST and KMNIST in this regime. In other words, even when trained on inputs that appear almost devoid of recognizable structure, the networks retain predictive power substantially higher than random chance. In addition, we compare the empirical results to the centroid baseline on the clean data, as predicted by Eq. (15), finding good agreement near  $p \sim 0.9$ . In actuality, the predicted centroid-classification rule competes with a noisy background that undermines test accuracy; this will be discussed in detail in Sec. III.

We also note an interesting feature of the standard deviation of the test accuracy as a function of  $p$  in Fig. 1: It is nonmonotonic. Specifically, the standard deviation is relatively small near  $p = 1$ , increases as  $p$  decreases, reaches a maximum at an intermediate corruption level, and then decreases again for smaller  $p$ . At first glance, this is surprising: A reasonable expectation is for the network to be maximally uncertain at  $p = 1$ , where it is trained on pure noise. However, in Appendix I we show that for multiclass problems ( $C > 2$ ), this is not the case: The standard deviation generically peaks at  $p^* < 1$ . This effect arises from the indicator form of the test accuracy.

## C. Robustness to simultaneous training-set and testing-set corruption

So far, we have trained models with some corruption level  $p$  and tested them on clean data. We now consider a generalized scenario, where *both* the training set and the test set are corrupted, with corruption levels  $p_{\text{train}}$  and  $p_{\text{test}}$ , respectively. Figure 3 shows the resulting test accuracy as a function of  $p_{\text{train}}$ , for several values of  $p_{\text{test}}$  (we emphasize that the  $p_{\text{test}} = 0$  curve is the same as in Fig. 1).

The naive expectation might be that the optimal value of  $p_{\text{train}}$  would be zero: One would expect that training with clean data will always be preferred. However, we find that for a given value of  $p_{\text{test}}$ , the behavior is nonmonotonic as a function of  $p_{\text{train}}$ , and in fact the optimal value is  $p_{\text{train}} \simeq p_{\text{test}}$ . The interpretation is that, when training and testing use the same level of corruption, the model is trying to generalize within the same data distribution, rather than out of distribution, which is more likely to succeed. In contrast, for a fixed value of  $p_{\text{train}}$ , it is always optimal to use  $p_{\text{test}} = 0$ , as is evident from Fig. 3. In Appendix B, we demonstrate this and also discuss a strategy to handle situations when  $p_{\text{test}}$  is not known exactly.

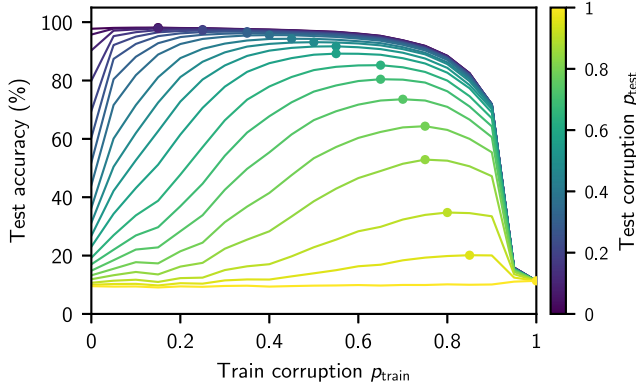


FIG. 3. Train-time and test-time corruption in MNIST data: Accuracy vs train corruption level  $p_{\text{train}}$ , for several values of test corruption  $p_{\text{test}}$ . We observe a nonmonotonic behavior, where the optimal  $p_{\text{train}}$  (dots in the figure) is near  $p_{\text{test}}$ , rather than 0. All networks and hyperparameters are the same as in Fig. 1.

### III. ANALYTICAL TREATMENT

We now turn to an analytical explanation of the empirical robustness observed above. We show that, in the severe-corruption regime, the network output simplifies dramatically: Rather than learning a complicated denoising rule, a wide MLP trained on corrupted inputs reduces at leading order to a centroid classifier on the clean data. We derive this result using the NTK description of infinite-width networks, in which training with mean-squared error (MSE) is equivalent to kernel regression with the NTK.

Throughout this section, we use the standard NTK initialization, with weights and biases drawn independently from mean-zero Gaussian distributions [37]. We also normalize both the clean training and test inputs before applying corruption. Specifically, each input is normalized and centered so that

$$\frac{1}{d} \sum_{k=1}^d x_{k\alpha} = 0, \quad \frac{1}{d} \sum_{k=1}^d x_{k\alpha}^2 = 1, \quad (3)$$

where  $\alpha = \chi$  denotes a training example and  $\alpha = *$  denotes a test example. This normalization is applied to both the training and test sets for notational and conceptual simplicity; the more general case is discussed in Appendix C. Since the clean data are centered and normalized in this section, we correspondingly take the replacement noise to be

$$u_{k\chi} \sim \text{Unif}[-\sqrt{3}, \sqrt{3}],$$

so that the pure-noise inputs also have mean zero and variance one. This choice aligns the first two moments of the noise with the normalized data.

#### A. Derivation of the high-noise centroid rule

We now present the effective decision rule learned in the high-noise regime. In the infinite-width limit, training an  $L$ -

layer MLP<sup>1</sup> with MSE loss is equivalent to kernel regression with the NTK [26]. We will go through a sketch of the derivation in this section; a detailed derivation for arbitrary depth and activation is provided in Appendix C.

The class- $i$  output logit of the MLP on a clean test point  $\mathbf{x}_*$  is given by [38]

$$\tilde{f}_i(\mathbf{x}_*; \epsilon) = \tilde{\Theta}_{*\chi}^{(L)} [\tilde{\Theta}_{\chi\chi}^{(L)}]^{-1} \mathbf{y}_i, \quad (4)$$

where  $\tilde{\Theta}_{\chi\chi}^{(L)}$  is the depth- $L$  NTK on the noisy training set,  $\tilde{\Theta}_{*\chi}^{(L)}$  is the test-train NTK,  $\mathbf{y}_i \in \{0, 1\}^N$  is the target one-hot vector for class  $i$ , and we have defined the signal strength

$$\epsilon \equiv 1 - p \quad (5)$$

for convenience. The pure-noise limit corresponds to  $\epsilon = 0$ , while  $\epsilon = 1$  corresponds to clean training inputs. Notice that the test-train NTK is a  $1 \times N$  vector with respect to the training data, whereas the train-train NTK is an  $N \times N$  matrix in the same space.

We now sketch the derivation of the logit output in the high-noise limit. The key simplification occurs in the first layer, where the NTK has the simple form

$$\tilde{\Theta}_{\alpha\beta}^{(1)}(\epsilon) = C_b + C_w \tilde{X}_{\alpha\beta}(\epsilon), \quad (6)$$

where  $C_w > 0$  and  $C_b > 0$  are the weight and bias variances at initialization,

$$\tilde{X}_{\alpha\beta}(\epsilon) \equiv \frac{1}{d} \sum_{k=1}^d \tilde{x}_{k\alpha} \tilde{x}_{k\beta} \quad (7)$$

denotes the normalized noisy feature overlap matrix, and  $\alpha, \beta$  denote either train or test indices. In a mean-field-inspired decomposition, we separate the overlap matrix into a deterministic term obtained from averaging over the corruption noise, plus a subleading fluctuation

$$\tilde{X}_{\alpha\beta}(\epsilon) = X_{\alpha\beta}(\epsilon) + \delta \tilde{X}_{\alpha\beta}(\epsilon), \quad (8)$$

where

$$X_{\alpha\beta}(\epsilon) \equiv \mathbb{E}[\tilde{X}_{\alpha\beta}(\epsilon)] \quad (9)$$

is the noise-averaged overlap and

$$\delta \tilde{X}_{\alpha\beta}(\epsilon) = \frac{1}{\sqrt{d}} \eta_{\alpha\beta}(\epsilon) + \mathcal{O}(d^{-1}) \quad (10)$$

is a nondeterministic fluctuation of  $\mathcal{O}(d^{-1/2})$ . In the large- $d$  limit,  $\eta_{\alpha\beta}(\epsilon)$  is asymptotically Gaussian with zero mean via the central limit theorem, while the  $\mathcal{O}(d^{-1})$  term captures the subleading finite- $d$  corrections to this Gaussian limit.

Through the layer-to-layer recursion formulas [37,38], the depth- $L$  NTK inherits its entire data dependence, and, in particular, the mean-plus-fluctuation split of Eq. (8), from  $\tilde{X}_{\alpha\beta}$ . The full layerwise propagation is carried out in Appendix C; here, we simply track the two pieces of Eq. (8) through to the output.

We work near the pure-noise point and in the large-feature-dimension limit:  $\epsilon \ll 1$  and  $\sqrt{d} \gg 1$ . Throughout this

<sup>1</sup>We use the convention that a depth  $L$  network has  $L - 1$  hidden layers.

expansion, the argument (0) denotes evaluation at pure noise,  $\epsilon = 0$  (equivalently,  $p = 1$ ). With this in mind, we perform a joint first-order expansion of the noisy overlap matrix [Eq. (8)] in  $\epsilon$  and  $1/\sqrt{d}$ , leading to

$$\tilde{X}_{\alpha\beta}(\epsilon) = X_{\alpha\beta}(0) + \epsilon \partial_{\epsilon} X_{\alpha\beta}(0) + \frac{1}{\sqrt{d}} \eta_{\alpha\beta}(0) + \mathcal{O}(\epsilon^2) + \mathcal{O}(\epsilon d^{-1/2}) + \mathcal{O}(d^{-1}), \quad (11)$$

where we have also expanded the nondeterministic fluctuation  $\eta_{\alpha\beta}$  about  $\epsilon = 0$ .

Feeding this structure through the recursion, the depth- $L$  NTK takes on a similar form,

$$\tilde{\Theta}_{\alpha\beta}^{(L)}(\epsilon) = \Theta_{\alpha\beta}^{(L)}(0) + \epsilon \partial_{\epsilon} \Theta_{\alpha\beta}^{(L)}(0) + \frac{1}{\sqrt{d}} \eta_{\alpha\beta}^{(L)}(0) + \mathcal{O}(\epsilon^2) + \mathcal{O}(\epsilon d^{-1/2}) + \mathcal{O}(d^{-1}), \quad (12)$$

where  $\Theta_{\alpha\beta}^{(L)}(0)$  is the depth- $L$  NTK built from the pure-noise averaged overlap  $X_{\alpha\beta}(0)$  and  $\eta_{\alpha\beta}^{(L)}(0)$  is zero-mean Gaussian fluctuation inherited from  $\delta\tilde{X}$ .

Substituting Eq. (12) into Eq. (4) and expanding to leading order in  $\epsilon$  and  $d^{-1/2}$ , we obtain

$$f_i(\mathbf{x}_*; \epsilon) \approx C^{(L)} + A^{(L)} \sqrt{\frac{N/C}{d}} \hat{\xi}_i + B^{(L)} \frac{\epsilon N}{C} \left[ \frac{1}{d} \sum_{k=1}^d x_{k*} (\bar{x}_k^i - \bar{x}_k) \right], \quad (13)$$

where  $A^{(L)}$ ,  $B^{(L)}$ , and  $C^{(L)}$  are class-independent constants that encode depth, activation, initialization, and details of the noise distribution;  $\hat{\xi}_i$  is a class-dependent Gaussian fluctuation with zero mean and unit variance, and

$$\bar{x}_k^i = \frac{C}{N} \sum_{\chi \in i} x_{k\chi}, \quad \bar{x}_k = \frac{1}{N} \sum_{\chi} x_{k\chi} \quad (14)$$

are the centered empirical mean, or centroid, of class  $i$  and the global empirical mean, respectively, in feature  $k$  for the *clean* training data. The crucial point is that the leading class-dependent signal depends on the training data only through these *clean* centered-class means. Despite being trained on corrupted inputs, the network extracts from the noisy data a class-dependent statistic depending only on the clean dataset. At this order, the noisy part of the inputs contributes only the fluctuation term  $\hat{\xi}_i$ , while the discriminating signal is the overlap between the test point and the clean class centroid.

Therefore, the leading-order decision rule learned by the network in the joint limit  $\epsilon \ll 1$  and  $\sqrt{d} \gg 1$  is

$$i^* = \arg \max_{i=1,\dots,C} \frac{1}{d} \sum_{k=1}^d x_{k*} (\bar{x}_k^i - \bar{x}_k). \quad (15)$$

In other words, the NTK calculation predicts that, in the high-noise regime, the network reduces to a nearest-class-mean, or centroid, classifier [27] with an inner-product score in feature space. Remarkably, the leading-order decision rule is universal across depth, activation, and noise distribution. Despite the depth and nonlinearity of the original model, classifica-

tion collapses to simple linear template matching against one prototype per class.

We note that the coefficient  $B^{(L)}$  multiplying the centroid term in Eq. (13) is guaranteed to be non-negative under our normalization convention [Eq. (3)] and unit-variance noise models (see Appendix D for further details). However, non-negativity of  $B^{(L)}$  is not guaranteed for arbitrary non-normalized data or completely general noise models. For settings where  $B^{(L)} < 0$ , the leading decision rule is simply obtained by reversing the orientation of the centroid score, equivalently, by replacing  $\arg \max$  with  $\arg \min$ . Thus, the leading class-dependent response remains a centroid-overlap rule, with the sign of  $B^{(L)}$  determining its orientation.

Finally, we comment on the controllability of the approximation at large training dataset size  $N$ . The prefactor of the centroid score in Eq. (13) comes from expanding the inverse NTK around the pure-noise kernel. For moderate  $N$ , this expansion is controlled by the small signal strength  $\epsilon$ . At larger  $N$ , higher-order terms in  $\epsilon$  can become important; however, these terms primarily renormalize the overall coefficient of the centroid statistic, leaving the leading decision rule unchanged (further details are given in Appendix C). The full expansion obtained by substituting higher-order NTK corrections into Eq. (4) does not necessarily converge uniformly in  $N$ . Nonetheless, as shown in the next subsection, the leading correction already agrees well with the empirical network outputs and test accuracy in the high-noise regime.

## B. Comparison with finite-width networks

We now compare the theoretical prediction for the logit output [Eq. (13)] to finite-width networks in a controlled empirical setting matched to the assumptions of the high-noise NTK calculation. This setting differs from the motivating experiments in Fig. 1, which use finite-width ReLU MLPs, cross-entropy loss, and raw image inputs in  $[0, 1]$  to establish the robustness phenomenon in a standard classification setup. Here, in contrast, we use the normalization convention of Eq. (3) for the inputs, MSE loss, wider networks with erf activation, and fewer training data points so that the numerical experiment isolates the leading-order mechanism predicted by the theory. The theoretical prediction is not intended to reproduce every implementation detail of Fig. 1; rather, it identifies a tractable high-noise mechanism, which we validate directly in the finite-width experiments of Fig. 4. Further experimental details on the empirical results presented here can be found in Table I in Appendix A.

Restoring the corruption probability  $p = 1 - \epsilon$ , the form of Eq. (13) motivates the effective model

$$f_i(\mathbf{x}_*) = a(1-p) \left[ \frac{1}{d} \sum_{k=1}^d x_{k*} (\bar{x}_k^i - \bar{x}_k) \right] + b + c \hat{\xi}_i, \quad (16)$$

where  $a$ ,  $b$ , and  $c$  absorb the renormalized architecture-dependent prefactors and fluctuation scale. For simplicity, we model the fluctuations  $\hat{\xi}_i$  as i.i.d. standard normal random variables. To estimate these coefficients, we train  $m$  MLPs for each of  $M$  independent noisy-dataset realizations, for a total of  $mM$  trained networks. For each test point and class, we first

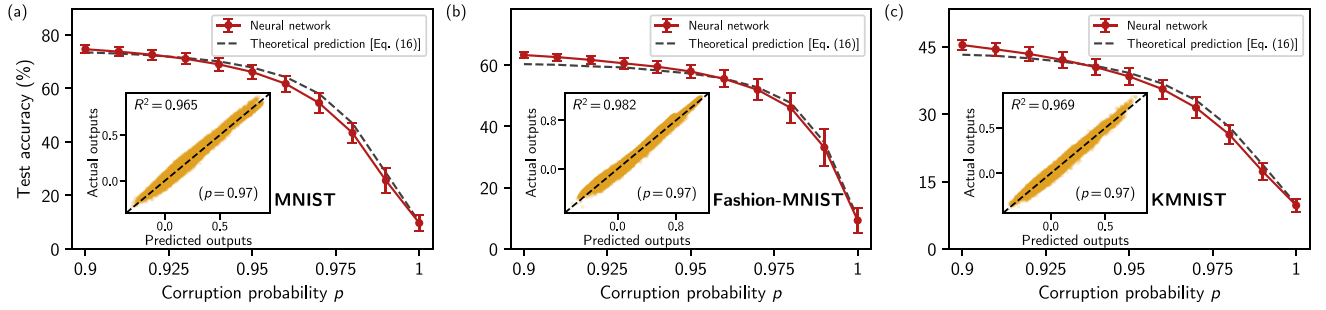


FIG. 4. Numerical verification of the high-noise theory. All networks used to create the data in this figure are width-2048, three-layer erf MLPs, trained with MSE loss on noisy (a) MNIST, (b) Fashion-MNIST, and (c) KMNIST with  $N = 4000$ . Main panels: Mean clean test accuracy vs replacement-noise strength  $p$  for a nested ensemble of 20 noisy training sets and ten MLPs per noisy set (red curve). Error bars show one standard deviation. The gray-dashed curve is the mean clean test accuracy of the fitted effective model in Eq. (16). Insets: Output-level comparison at  $p = 0.97$ , using a nested ensemble of 100 noisy training sets and 50 MLPs per noisy set. Each point corresponds to one clean test image and one candidate class, comparing the ensemble-mean network output with the fitted prediction from Eq. (16). The black dashed line in the inset indicates perfect agreement.

average the output logit over the  $m$  networks corresponding to a fixed noisy realization. This produces  $M$  realization-averaged logits per class, which are the empirical objects to be compared with Eq. (16). We then average these logits over the  $M$  noisy realizations and determine  $a$  and  $b$  by a least-squares fit over all test points and classes to the mean of Eq. (16); the coefficient  $c$  does not enter this step because it multiplies a zero-mean fluctuation. Finally, we set  $c$  equal to the standard deviation of the  $M$  realization-averaged logits, pooled over all test points and classes. Importantly, this is a global three-parameter fit: We do *not* fit separate coefficients for different test examples.

Figure 4 tests this picture in two complementary ways. The insets compare the empirical mean output logits of the MLPs on the full MNIST, Fashion-MNIST, and KMNIST test sets with the prediction obtained from the mean of Eq. (16). After fitting the two global coefficients  $a$  and  $b$ , we find nearly exact agreement for each dataset. This provides strong evidence that, after averaging over noise realizations, the outputs of the MLP collapse onto the mean of the centroid model, consistent with centroid-classifier behavior. The main panels compare the clean test accuracy of the empirical ensemble with that of the full centroid model in Eq. (16), now including the fluctuation term controlled by the fitted coefficient  $c$ . The agreement for each dataset shows that the centroid model captures not only the mean output structure, but also the fluctuations that govern classification accuracy. In particular, it accurately reproduces the decay of the test accuracy as the noise probability  $p$  increases. The main panels also indicate where the empirical networks begin to depart from the leading-order centroid description, signaling the onset of higher-order corrections in  $(1 - p)$ . In our data, this deviation becomes noticeable for  $p \lesssim 0.9$ . We provide additional tests in Appendixes F and G.

The agreement in Fig. 4 should therefore be interpreted as a controlled validation of the high-noise centroid mechanism, rather than as a claim that the theory reproduces every implementation detail of the raw-input cross-entropy experiments in Fig. 1. Nevertheless, it indicates that the same centroid mechanism remains visible in finite-width networks once the

experimental setting is chosen to match the assumptions of the calculation.

### C. Signal-to-noise ratio

Equation (13) for the output logit also implies a scaling law, since it predicts that, at least for intermediate  $N$ , the class-dependent mean grows as  $(1 - p)(N/C)$ , while the standard deviation scales as  $\sqrt{(N/C)/d}$ . Therefore, we expect the signal-to-noise ratio (SNR) to scale as

$$\text{SNR} \sim (1 - p)^2 d \frac{N}{C}. \quad (17)$$

The theory therefore predicts improved classification accuracy as the surviving signal fraction  $1 - p$ , the feature dimension  $d$ , or the number of training examples per class  $N/C$  are increased. We test the  $N, d$  dependence on MNIST at fixed replacement-noise probability  $p = 0.97$  and class count  $C = 10$ . With these quantities held fixed, Eq. (17) predicts  $\text{SNR} \propto Nd$ , corresponding to a unit slope on logarithmic axes. Figure 5(a) directly tests this prediction using the empirical SNR of the true-class output logit. Averaging  $\ln(\text{SNR})$  over randomly sampled clean test examples, the results for different  $N$  and  $d$  approximately collapse onto a linear trend in  $\ln(Nd)$ . The fitted slope of 1.03 is close to the predicted value of unity, supporting the proposed scaling. Figure 5(b) shows the accompanying improvement in clean test accuracy with increasing  $N$  and  $d$ .

The analysis in this section has focused on uniform replacement noise for concreteness. However, the centroid-classification result is not tied to that specific choice of corruption model. Rather, for the two broad classes of input noise most relevant in practice—general replacement noise and general additive noise—the same leading centroid structure persists. In Appendix E, we present analogous results for the additive noise model [Eq. (2)], demonstrating the generality of our findings.

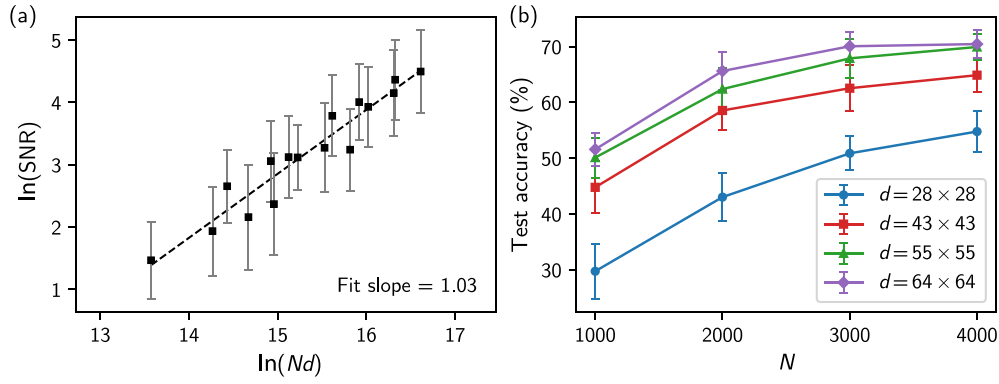


FIG. 5. Empirical  $N, d$  scaling at fixed replacement-noise probability  $p = 0.97$  and number of classes  $C = 10$  for width-2048, three-layer erf MLPs trained with MSE loss on MNIST. (a) Mean  $\ln(\text{SNR})$  vs  $\ln(Nd)$  over 500 randomly sampled clean test examples. For each example, true-class logits are first averaged over ten independently initialized networks trained on each noisy training dataset. The SNR is the squared mean of these averaged logits divided by their sample variance across 20 noisy training datasets. Squares and error bars show the mean and one standard deviation of  $\ln(\text{SNR})$  across test examples; the dashed line is a linear fit to the means. (b) Mean clean test accuracy vs  $N$  at fixed  $d$ , using predictions from initialization-averaged logits. Means and one-standard-deviation error bars are computed across the 20 noisy training datasets. Larger feature dimensions  $d$  are obtained by resizing MNIST images using bilinear interpolation.

#### D. Permutation symmetry and the centroid response

The centroid rule [Eq. (15)] is independent of noise distribution, depth, and activation. Here, we give intuition for this universality within the NTK framework using a linear-response argument: we treat the trained network as a system sitting at a permutation-symmetric pure-noise reference state and view the surviving signal as a small perturbation away from it. This provides a symmetry interpretation of the high-noise expansion derived in Appendix C. It should be read as a minimal linear-response model rather than a second microscopic derivation of the corruption channels of Sec. II, but it is sufficient to extract the leading-order class-dependent decision rule [Eq. (15)].

After separating the  $\mathcal{O}(d^{-1/2})$  stochastic fluctuations, the deterministic part of the noisy NTK depends on the data only through the noise-averaged overlap [Eq. (9)]. At pure noise,  $\epsilon = 1 - p = 0$ , the training examples become exchangeable: The noise-averaged train-train overlap is invariant under any relabeling  $\pi \in S_N$  of the training examples,

$$X_{\pi(\alpha)\pi(\beta)}(0) = X_{\alpha\beta}(0). \quad (18)$$

That is,  $X_{\alpha\beta}(0)$  depends on  $\alpha, \beta$  only through whether  $\alpha = \beta$ . This symmetry structure is guaranteed at pure noise for any distribution that is i.i.d. with respect to data and feature space and has bounded mean and variance.

Motivated by this structure, we introduce a minimal pure-noise reference ensemble for the training set. This reference ensemble consists of featureless, i.i.d. random variables  $\{\xi_{k\alpha}\}$ , whose purpose is to capture the exchangeable structure of the deterministic part of the noisy NTK at the pure-noise fixed point. The corresponding pure-noise-averaged test-train and train-train overlap matrices are

$$X_{0,*\alpha} = \frac{1}{d} \sum_{k=1}^d x_{k*} \mathbb{E}[\xi_{k\alpha}], \quad (19)$$

$$X_{0,\alpha\beta} = \frac{1}{d} \sum_{k=1}^d \mathbb{E}[\xi_{k\alpha} \xi_{k\beta}], \quad (20)$$

where the expectation  $\mathbb{E}[\cdot]$  is taken with respect to the noise distribution. We leave the first two moments of the reference noise arbitrary, assuming only that they are bounded. For later convenience, we assume the first moment vanishes and denote the second moment by  $v$ .<sup>2</sup> Adopting the normalization convention of Eq. (3) for the clean training inputs  $\{x_{k\alpha}\}$  and test input  $x_{k*}$ , the imposed mean centering makes the pure-noise test-train overlap vanish,  $X_{0,*\alpha} = 0$ , while the pure-noise train-train overlap remains diagonal  $X_{0,\alpha\beta} = \delta_{\alpha\beta} v$ .

We treat the network trained on the pure-noise ensemble  $\{\xi_{k\alpha}\}$  as a pure-noise fixed point. Now, suppose we perturb the pure-noise inputs by a small signal proportional to the clean features,

$$\xi_{k\alpha} \rightarrow \xi_{k\alpha} + \epsilon x_{k\alpha}. \quad (21)$$

To study the response, let  $\delta$  denote the leading-order variation along the perturbation path  $z_{k\alpha}(\epsilon) = \xi_{k\alpha} + \epsilon x_{k\alpha}$ . That is, for any function  $F$  built from the training inputs, define

$$\delta F \equiv \epsilon \frac{d}{d\epsilon} \bigg|_{\epsilon=0} F[z(\epsilon)], \quad (22)$$

so that  $\delta F = \mathcal{O}(\epsilon)$ . The leading-order response of the test-train overlap is

$$\delta X_{0,*\alpha} = \frac{\epsilon}{d} \sum_{k=1}^d x_{k*} x_{k\alpha}, \quad (23)$$

<sup>2</sup>The value of the first moment is inconsequential for the rest of the calculation; a nonzero first moment simply renormalizes the constants  $r^{(\ell)}$ ,  $s^{(\ell)}$ , and  $t^{(\ell)}$  introduced later in the subsection. In contrast, the value of the second moment can affect the orientation of the centroid rule, as discussed below.

while the response of the train-train overlap vanishes, since its leading nonzero term is  $\mathcal{O}(\epsilon^2)$ . Thus, only  $\delta X_{0,*\alpha}$  carries the signal at leading order.<sup>3</sup>

We now compute the response at arbitrary depth  $L$ . Let  $\Theta_{0,\alpha\beta}^{(\ell)}$  and  $\Theta_{0,*\alpha}^{(\ell)}$  denote the depth- $\ell$  train-train and test-train NTKs at pure noise, respectively. At leading order in  $\epsilon$ , the class- $i$  logit responds as

$$\delta f_i(\mathbf{x}_*) = \sum_{\alpha,\beta=1}^N \delta \Theta_{0,*\alpha}^{(L)} [\Theta_0^{(L)}]_{\alpha\beta}^{-1} y_{i\beta}, \quad (24)$$

where the variation of the inverse train-train NTK vanishes at leading order in this minimal treatment. Two objects therefore determine the response: the inverse pure-noise train-train NTK  $[\Theta_0^{(L)}]^{-1}$  and the pure-noise test-train NTK variation  $\delta \Theta_{0,*\alpha}^{(L)}$ . Both follow from applying the layer-to-layer recursion relations [38] to the pure-noise fixed point:

$$K_{0,\alpha\beta}^{(\ell+1)} = C_b + C_w \langle \sigma_\alpha \sigma_\beta \rangle_{K_0^{(\ell)}}, \quad (25)$$

$$\Theta_{0,\alpha\beta}^{(\ell+1)} = K_{0,\alpha\beta}^{(\ell+1)} + C_w \langle \sigma'_\alpha \sigma'_\beta \rangle_{K_0^{(\ell)}} \Theta_{0,\alpha\beta}^{(\ell)}, \quad (26)$$

where  $K_0^{(\ell)}$  denotes the depth- $\ell$  pure-noise preactivation kernel, and the layer-1 initial condition is fixed directly by the pure-noise overlap,

$$K_{0,\alpha\beta}^{(1)} = \Theta_{0,\alpha\beta}^{(1)} = C_b + C_w X_{0,\alpha\beta}. \quad (27)$$

In the recursion relations,  $\sigma_\alpha \equiv \sigma(u_\alpha)$ ,  $\sigma'_\alpha$  is its derivative, and  $\langle \cdots \rangle_K$  is an expectation over the zero-mean bivariate Gaussian preactivations  $(u_\alpha, u_\beta)$  with  $2 \times 2$  covariance  $K$  [Eq. (C2)].

Consider first the inverse train-train NTK. At pure noise, the train-train overlap is permutation invariant, and hence so is the layer-1 NTK,  $\Theta_{0,\alpha\beta}^{(1)} = r^{(1)} \delta_{\alpha\beta} + s^{(1)}$ , where  $r^{(1)} = C_w v$  and  $s^{(1)} = C_b$ . The recursions at pure noise are permutation equivariant; if the depth-1 kernel is permutation invariant, then this invariance is preserved at arbitrary depth  $\ell$ . Therefore, the depth- $L$  train-train NTK therefore retains the two-parameter form  $\Theta_{0,\alpha\beta}^{(L)} = r^{(L)} \delta_{\alpha\beta} + s^{(L)}$ , for constants  $r^{(L)}$  and  $s^{(L)}$  that depend on depth, activation, initialization, and the moments of the noise distribution. Equivalently, the depth- $L$  NTK at pure noise lives in the projector algebra  $\text{span}(\mathbf{P}_\parallel, \mathbf{P}_\perp)$ , with  $\mathbf{P}_\parallel = \frac{1}{N} \mathbf{1}\mathbf{1}^T$  the projector onto the permutation-invariant sector and  $\mathbf{P}_\perp = \mathbf{I} - \mathbf{P}_\parallel$  its complement:

$$\Theta_0^{(L)} = r^{(L)} \mathbf{P}_\perp + (r^{(L)} + s^{(L)} N) \mathbf{P}_\parallel. \quad (28)$$

The NTK is an example of a Gram matrix, which is strictly positive definite for distinct inputs [26]. Therefore, its eigenvalues  $r^{(L)} > 0$  and  $r^{(L)} + s^{(L)} N > 0$  are strictly positive, and so it has a well-defined inverse

$$[\Theta_0^{(L)}]^{-1} = \frac{\mathbf{P}_\perp}{r^{(L)}} + \frac{\mathbf{P}_\parallel}{r^{(L)} + s^{(L)} N}. \quad (29)$$

<sup>3</sup>The train-train overlap vanishes at linear order in  $\epsilon$  under our convention that each training input has zero mean across features [see Eq. (3)].

Consider next the depth- $L$  test-train NTK variation  $\delta \Theta_{0,*\alpha}^{(L)}$ . The first-layer variation about pure noise is

$$\delta \Theta_{0,*\alpha}^{(1)} = \delta K_{0,*\alpha}^{(1)} = C_w \delta X_{0,*\alpha}. \quad (30)$$

Layer-to-layer propagation of the variation can be handled by differentiating the recursion relations using Price's theorem [39]. The resulting recursion relations for the test-train kernel and NTK variations about pure noise are

$$\delta K_{0,*\alpha}^{(\ell+1)} = C_w \langle \sigma'_* \sigma'_\alpha \rangle_{K_0^{(\ell)}} \delta K_{0,*\alpha}^{(\ell)}, \quad (31)$$

$$\begin{aligned} \delta \Theta_{0,*\alpha}^{(\ell+1)} &= \delta K_{0,*\alpha}^{(\ell+1)} + C_w \langle \sigma''_* \sigma''_\alpha \rangle_{K_0^{(\ell)}} \Theta_{0,*\alpha}^{(\ell)} \delta K_{0,*\alpha}^{(\ell)} \\ &\quad + C_w \langle \sigma'_* \sigma'_\alpha \rangle_{K_0^{(\ell)}} \delta \Theta_{0,*\alpha}^{(\ell)}, \end{aligned} \quad (32)$$

where we have used the fact that diagonal variations of the test-train kernel and NTK vanish at leading order.

We now show how permutation symmetry in the training index  $\alpha$  reduces layer and activation dependence to an overall scaling of the first-layer variation. To see this, consider the entries of the layer-1 test-train kernel at pure noise, which are permutation invariant in training input  $\alpha$

$$K_{0,**}^{(1)} = C_b + C_w, \quad K_{0,\alpha\alpha}^{(1)} = C_b + C_w v, \quad K_{0,*\alpha}^{(1)} = C_b, \quad (33)$$

where  $v$  is the second moment of the noise distribution. Permutation equivariance of the pure-noise recursion [Eq. (25)] preserves this  $\alpha$  independence to arbitrary depth. Consequently, the pure-noise activation-dependent expectation values and NTK entries appearing in Eqs. (31) and (32) are independent of  $\alpha$ . Since  $\delta \Theta_{0,*\alpha}^{(1)} \propto \delta X_{0,*\alpha}$ , we conclude that the recursion merely rescales the pure-noise overlap variation by a constant:

$$\delta \Theta_{0,*\alpha}^{(L)} = t^{(L)} \delta X_{0,*\alpha}, \quad (34)$$

where  $t^{(L)}$  depends on depth, activation, initialization, and the moments of the noise distribution.

Substituting Eqs. (29) and (34) into Eq. (24), and using the class-balanced label contractions

$$\mathbf{P}_\perp \mathbf{y}_i = \mathbf{y}_i - \frac{1}{C} \mathbf{1}, \quad \mathbf{P}_\parallel \mathbf{y}_i = \frac{1}{C} \mathbf{1}, \quad (35)$$

the permutation-invariant sector  $\mathbf{P}_\parallel$  contributes only a class-independent constant, while the  $\mathbf{P}_\perp$  sector carries the class signal. With  $\delta X_{0,*\alpha}$  from Eq. (23), this gives

$$\begin{aligned} \delta f_i(\mathbf{x}_*) &= \frac{t^{(L)} \epsilon N}{r^{(L)} C} \left[ \frac{1}{d} \sum_{k=1}^d x_{k*} (\bar{x}_k^i - \bar{x}_k) \right] \\ &\quad + \text{class-independent terms}, \end{aligned} \quad (36)$$

which is exactly the centroid rule. The constants  $t^{(L)}$  and  $r^{(L)}$  encode depth, activation, initialization, and the first two moments of the noise distribution, while the class-dependent structure is fixed by permutation symmetry at the pure-noise fixed point.

A remark on orientation is useful. The prefactor  $t^{(L)}/r^{(L)}$  is common to all classes, so only its sign matters: A positive sign gives the nearest-class-mean rule, while a negative sign reverses the orientation of the centroid score. Since  $r^{(L)} > 0$ , the orientation is determined by the sign of  $t^{(L)}$ . Under the

unit-variance convention  $v = 1$ , the layer-1 test-train kernel entries become

$$K_{0,**}^{(1)} = K_{0,\alpha\alpha}^{(1)}|_{v=1} = C_b + C_w, \quad K_{0,*\alpha}^{(1)} = C_b, \quad (37)$$

which can be determined using Eq. (33). Thus, the test and training diagonals coincide, and the off-diagonal entry is non-negative. These properties imply that the activation-dependent transfer factors  $\langle h_* h_\alpha \rangle_{K_0^{(l)}}$ , with  $h$  being any of  $\sigma, \sigma', \sigma''$ , appearing in the kernel and NTK variation recursions are non-negative at every depth for any activation function—We show this in detail in Appendix D. Consequently, for unit-variance noise under the normalization convention [Eq. (3)], we have  $t^{(L)} \geq 0$ .

For general  $v \neq 1$  or non-normalized data, the test and training diagonals of the pure-noise test-train kernel need not coincide, and the sign is not guaranteed for every activation. If the prefactor is negative, however, the leading decision rule is obtained by reversing the orientation of the centroid score, equivalently, by replacing  $\arg \max$  with  $\arg \min$ . Thus, the leading class-dependent response remains a centroid-overlap rule for any depth, activation, and i.i.d. noise distribution, with the sign of  $t^{(L)}$  fixing the orientation.

#### IV. DISCUSSION AND OUTLOOK

The problem of training neural networks with heavily corrupted inputs has revealed numerous surprises. First, we found that MLPs are remarkably robust to attribute noise, capable of classifying heavily corrupted images with accuracy much higher than chance. This phenomenon, observed across several datasets, prompted us to tackle the problem analytically, with the naive expectation of revealing the complicated “denoising” process employed by the networks, enabling them to perform so well. Using a controlled expansion of the NTK in the strong-corruption regime, we found that in fact the networks learn a very simple rule: They compare each test image to the average of all training images of a certain class and check which one is the most similar. This nearest-class-mean (centroid) rule, which is the dominant contribution in the strong-corruption limit, emerges despite the complexity and nonlinearity enabled by the network. The analytical result also leads to an effective signal-to-noise ratio [Eq. (17)], which reveals that performance improves with feature dimension and dataset size; we have confirmed that empirically as well. Taken together, these results show that, in the high-attribute-noise regime, wide MLPs have a universal leading class-dependent response: a centered centroid classifier. The finite-width networks studied here visibly follow this mechanism, explaining their surprising robustness on the MNIST-family datasets.

The symmetry interpretation in Sec. III D explains why this simple centroid rule is not a coincidence of the first-layer calculation: It is fixed by the exchangeability of the pure-noise kernel and by the permutation equivariance of the NTK recursion. This observation also suggests a broader interpretation of the formalism. The essential ingredients are not the particular MLP recursion alone; the same reasoning should apply beyond fully connected MLPs to any architecture with a well-defined kernel limit whose pure-noise kernel has the same exchangeability structure. When those ingredients are present,

linear response around the high-noise fixed point identifies the first class-dependent correction to the predictor; in the present setting, that correction is precisely the centroid rule. More broadly, this points to a way of understanding learning phenomena by locating tractable fixed points of a network’s effective description and deriving the leading decision-rule corrections that appear when a small signal breaks the fixed-point symmetry.

The success of the NTK prediction for finite-width networks suggests that the high-noise regime has an effective universality beyond the exactly solvable infinite-width limit. Although finite width, nonlinear feature learning, and the choice of loss can change architecture-dependent prefactors and fluctuation scales, the dominant class-dependent direction appears to remain the centered centroid overlap. This should not be read as a claim that MLPs are robust to arbitrary training corruption on every classification dataset. Rather, the theory predicts a generic leading-order decision rule for MLPs in the high-noise regime, while the resulting accuracy depends on the geometry of the dataset itself. In the datasets studied here, MNIST, Fashion-MNIST, and KMNIST, the clean class centroids contain enough discriminative information to yield well-above-chance classification, which explains why the networks remain robust at high corruption in Fig. 1.

Finally, the analytical control enabled in the strong-corruption regime may also have broader implications. Our perturbative result in  $(1 - p)$  could, in principle, be extended to higher orders, thereby revealing progressively finer structure in the rules learned by the network. Systematically carrying out this expansion may provide access to the intermediate-corruption regime, or even suggest a controlled route toward the clean-data limit, using the corruption strength as an explicit expansion parameter. A related possible direction is the analysis of denoising models, including diffusion-type models, in the high-noise regime. In that setting, one would replace the classification target considered here by a regression target, such as the clean input. An analogous expansion around the pure-noise limit could then reveal what statistics of the clean data are recovered at leading order, and how progressively finer denoising information emerges at higher orders in the signal strength.

#### ACKNOWLEDGMENTS

We thank P. Frazier, T. Arias, C.-M. Jian, H. Pan, T. Nebabu, and J. Kim for useful discussions. J.T. acknowledges funding from the Center for Alkaline-Based Energy Solutions (CABES), part of the Energy Frontier Research Center (EFRC) program supported by the U.S. Department of Energy, under Grant No. DE-SC-0019445. A.B. was supported by a Cornell fellowship from the Cornell University Graduate School. H.K. acknowledges support by the NSF through the Grant No. OAC-2118310. O.L. acknowledges support from the Bethe-KIC postdoctoral fellowship at Cornell University.

#### DATA AVAILABILITY

There are no publicly available research data or software supporting this manuscript. Requests for further information or data should be sent to the authors.

## APPENDIX A: DETAILS OF THE EXPERIMENTS

The details of the numerical experiments presented in the main text and appendices are summarized in Table I. For each experiment, we specify the datasets and corruption models, network architectures, training hyperparameters, and evaluation procedures.

TABLE I. Summary of the experimental settings used to generate the results in Figs. 1 and 3–11, including datasets, corruption models, network architectures, training hyperparameters, and evaluation procedures.

| Figure  | Dataset and corruption                                                                                                                                                                                                                                                                                                                                                 | Model and training                                                                                                                                                                                                                                                                                                                                                                                                        |
|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Fig. 1  | Balanced MNIST, Fashion-MNIST, KMNIST; 54 210 training and 8920 clean test images each; replacement noise with $u \sim \text{Unif}[0, 1]$ , fixed across epochs; empirical $[0, 1]$ inputs with no per-example mean-shift or unit-RMS normalization.                                                                                                                   | Width-128, three-hidden-layer ReLU MLP; cross-entropy; Adam, learning rate $10^{-3}$ ; 20 epochs; batch size 128; 100 noisy training realizations per $p$ on a 21-point grid, denser above 0.8. Mean final-model accuracy with population standard deviation (SD). Horizontal clean-centroid references use inner-product classification on per-image centered, unit-RMS clean training and test data; no $p$ dependence. |
| Fig. 3  | MNIST; full 60 000 training and 10 000 test images; independent train/test replacement noise, $u \sim \text{Unif}[0, 1]$ ; raw $[0, 1]$ inputs. Training noise is resampled on each image access; one fixed test corruption per $p_{\text{test}}$ is shared across runs.                                                                                               | Same architecture, optimizer, epoch budget, batch size, and raw-input convention as Fig. 1; 20 training repeats per $p_{\text{train}}$ ; $p_{\text{train}} = 0, 0.05, \dots, 1$ and $p_{\text{test}} = 0, 0.0125, \dots, 1$ . Mean final-model accuracy; same experiment as Fig. 6.                                                                                                                                       |
| Fig. 4  | MNIST, Fashion-MNIST, KMNIST; balanced $N = 4000$ , $d = 784$ , and 10 000 clean test images each; replacement noise at $p = 0.90, \dots, 1.00$ in steps of 0.01. Clean train/test vectors are mean centered and scaled to centered RMS 1 before corruption; $u \sim \text{Unif}[-\sqrt{3}, \sqrt{3}]$ ; no postcorruption renormalization; noise fixed across epochs. | Width-2048, three-layer ( $L = 3$ ) erf MLP, counting the output layer; MSE loss; Adam, learning rate $10^{-4}$ ; 30 epochs; batch size 250. Accuracy: 20 noisy datasets, ten MLPs each; average logits within each dataset, then report mean accuracy and sample SD across datasets. Inset and calibration at $p = 0.97$ : 100 noisy datasets, 50 MLPs each; logits averaged over both ensemble levels.                  |
| Fig. 5  | MNIST with resized feature dimensions $28^2$ , $43^2$ , $55^2$ , and $64^2$ , using bilinear interpolation; replacement noise at $p = 0.97$ ; same class balancing, normalization and corruption conventions as Fig. 4.                                                                                                                                                | Width-2048, three-layer ( $L = 3$ ) erf MLP; MSE loss; Adam; 25 epochs; batch size 100; learning rate $10^{-4}$ ; $N \in \{1000, 2000, 3000, 4000\}$ ; 20 noisy datasets, ten MLPs each per $(N, d)$ . Accuracy uses initialization-averaged logits on all 10 000 clean test images; SNR uses the same 500 randomly sampled clean test images across $(N, d)$ .                                                           |
| Fig. 6  | Same MNIST dataset, raw-input convention, and independent train/test replacement corruption as Fig. 3.                                                                                                                                                                                                                                                                 | Same trained models and 20 repeats per $p_{\text{train}}$ as Fig. 3; 21 training and 81 test corruption values. AUC computed by trapezoidal integration over $p_{\text{test}}$ for each run; mean and standard error across runs.                                                                                                                                                                                         |
| Fig. 7  | MNIST, Fashion-MNIST, KMNIST; $N = 4000$ ; $p = 0.90, \dots, 1.00$ in steps of 0.01; same splits and clean-image normalization as Fig. 4. Noisy training inputs use $(1 - p)x + p\xi$ , $\xi \sim \mathcal{N}(0, I)$ , with no postcorruption renormalization; noise fixed across epochs; clean test sets.                                                             | Same width-2048, $L = 3$ erf MLP, MSE loss, Adam settings, 30 epochs, batch size 250, and evaluation as Fig. 4. Accuracy: 20 noisy datasets, ten MLPs each; inset and calibration at $p = 0.97$ : 100 noisy datasets, 50 MLPs each.                                                                                                                                                                                       |
| Fig. 8  | MNIST; same $(N, d)$ sweep and resizing as Fig. 5; additive-Gaussian noise at $p = 0.97$ ; same normalization and corruption conventions as Fig. 7.                                                                                                                                                                                                                    | Same width-2048, $L = 3$ erf MLP, MSE loss, Adam settings, 25 epochs, and batch size 100 as Fig. 5; 20 noisy datasets and ten MLPs each per $(N, d)$ . Same evaluation: 500 shared clean test images for SNR; all 10 000 for accuracy using initialization-averaged logits.                                                                                                                                               |
| Fig. 9  | MNIST, Fashion-MNIST, KMNIST; $N = 4000$ , $d = 784$ ; clean test sets. Same splits, normalization, and respective noise conventions as Figs. 4 and 7. Both grids contain $0, 0.1, \dots, 0.8$ , followed by $0.90, 0.91, \dots, 1.00$ for replacement and $0.85, 0.86, \dots, 1.00$ for additive Gaussian.                                                            | Width-2048, $L = 3$ erf MLP; MSE loss; Adam; 30 epochs; batch size 250; learning rate $10^{-4}$ ; 20 noisy datasets and ten MLPs each per $p$ . Evaluation matches the high-noise accuracy panels; their observations are reused. Effective model calibrated at $p = 0.97$ , not refitted at each $p$ .                                                                                                                   |
| Fig. 10 | MNIST, Fashion-MNIST, KMNIST; replacement and additive-Gaussian noise at $p = 0.90, 0.91, \dots, 1.00$ ; $N = 4000$ , $d = 784$ ; clean test sets. Same splits, normalization, and respective corruption conventions as Figs. 4 and 7.                                                                                                                                 | Width-2048, five-layer ( $L = 5$ ) ReLU MLP, counting the output layer; MSE loss; Adam; 30 epochs. Accuracy: 20 noisy datasets, ten MLPs each; insets and architecture-specific calibration at $p = 0.97$ : 100 noisy datasets, 50 MLPs each. Remaining training and evaluation settings match the depth-3 experiments.                                                                                                   |
| Fig. 11 | Mean-zero, unit-RMS normalized balanced subset of 4000 MNIST training examples; additive-Gaussian noise as in Fig. 7, fixed across epochs; 10 000 clean normalized test images.                                                                                                                                                                                        | Width-2048, three-layer ( $L = 3$ ) MLPs with erf, Gaussian Error Linear Unit (GELU), Swish, and Tanh; MSE loss; Adam; 20 epochs; batch size 250; learning rate $10^{-4}$ ; 50 noisy datasets with one initialization each per activation and $p = 0.90, \dots, 1.00$ in steps of 0.01; mean individual-model accuracy and sample SD.                                                                                     |

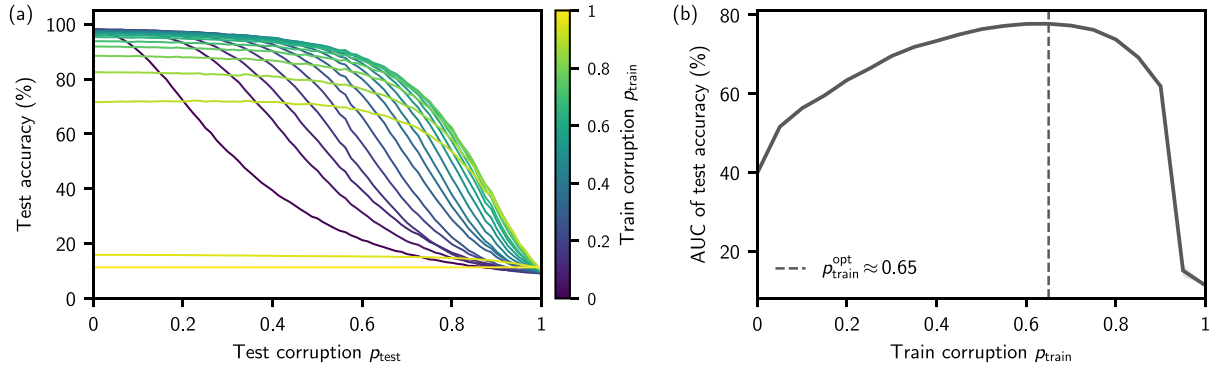


FIG. 6. (a) Test accuracy vs test corruption  $p_{\text{test}}$  for several values of the train corruption  $p_{\text{train}}$ . We find monotonically decreasing accuracy as a function of  $p_{\text{test}}$ , but for given  $p_{\text{test}}$  the behavior with respect to  $p_{\text{train}}$  is nonmonotonic (see also Fig. 3). (b) AUC of the test accuracy vs  $p_{\text{test}}$ , as a function of  $p_{\text{train}}$ . This is an overall measure of performance for a given  $p_{\text{train}}$ . We find that the AUC peaks around  $p_{\text{train}}^{\text{opt}} \approx 0.65$ . All network parameters are the same as in Fig. 3.

## APPENDIX B: FURTHER ANALYSIS OF SIMULTANEOUS TRAIN-TIME AND TEST-TIME CORRUPTION

In Sec. II, we showed results for train-time corruption as well as test-time corruption, characterized by corruption levels  $p_{\text{train}}$  and  $p_{\text{test}}$ , which are in general different. We found that for a given value of  $p_{\text{test}}$ , the optimal training corruption level is  $p_{\text{train}} \simeq p_{\text{test}}$ . Figure 6(a) shows the complementary view: accuracy as a function of  $p_{\text{test}}$  for different values of  $p_{\text{train}}$ .

There, we find monotonic behavior: It is always best to test on clean data ( $p_{\text{test}} = 0$ ), indicating an asymmetry in the effect of train-time and test-time corruption. Networks trained on corrupted data doing better when tested on clean data makes sense from the perspective of centroid classification: Testing on actual examples from the clean distribution is optimal for the nearest-mean-class classifier.

As an overall metric of performance of a network with given  $p_{\text{train}}$ , we calculate the area under the curve (AUC) of the test accuracy versus  $p_{\text{test}}$ . Figure 6(b) shows the AUC as a function of  $p_{\text{train}}$ , exhibiting nonmonotonic behavior with a peak around  $p_{\text{train}}^{\text{opt}} \approx 0.65$ . This implies that, if the test-time corruption is not known ahead of time, training with corruption level  $\approx 0.65$  will likely lead to optimal performance.

## APPENDIX C: HIGH-NOISE NTK EXPANSION

### 1. Setup

We consider a training set with  $N$  examples,  $C$  classes, and feature dimension  $d$ . The clean training features are denoted by  $\mathbf{x} \in \mathbb{R}^{d \times N}$ , while the corrupted training features are denoted by  $\tilde{\mathbf{x}} \in \mathbb{R}^{d \times N}$ . The clean labels are denoted by  $\mathbf{y} \in \mathbb{R}^{N \times C}$ , with  $\mathbf{y}_i \in \mathbb{R}^N$  the one-hot target vector for class  $i = 1, \dots, C$ . A clean test feature is denoted by  $x_* \in \mathbb{R}^d$ . The notation in this section is self-contained.

The index  $\chi$  is reserved for training examples. Other Greek indices, such as  $\alpha, \beta, \gamma, \delta$ , may refer either to training examples or to the test point  $*$ . We write  $z_\alpha$  for the input associated with index  $\alpha$ , so that  $z_\chi = \tilde{x}_\chi$  for a corrupted training input and  $z_* = x_*$  for the clean test input.

We work in the infinite-width NTK limit for an arbitrary  $L$ -layer MLP with activation function  $\sigma(\cdot)$ . The weights and biases at initialization are drawn i.i.d. from mean-zero Gaus-

sian distributions with variances  $C_w$  and  $C_b$ , respectively. In this limit, training with squared loss gives the class- $i$  output

$$f_i(x_*) = \tilde{\Theta}_{*\chi}^{(L)} [\tilde{\Theta}_{\chi\chi}^{(L)}]^{-1} \mathbf{y}_i, \quad (\text{C1})$$

where  $\tilde{\Theta}_{\chi\chi}^{(L)} \in \mathbb{R}^{N \times N}$  is the training-training NTK and  $\tilde{\Theta}_{*\chi}^{(L)} \in \mathbb{R}^{1 \times N}$  is the test-training NTK. The tilde indicates that the NTK is evaluated on the corrupted training features.

For arbitrary functions  $g$  and  $h$ , we define the two-dimensional Gaussian expectation by

$$\langle g_\alpha h_\beta \rangle_K = \frac{1}{2\pi \sqrt{\det K}} \int da db g(a) h(b) \times \exp \left[ -\frac{1}{2} \begin{pmatrix} a & b \end{pmatrix} \begin{pmatrix} K_{\alpha\alpha} & K_{\alpha\beta} \\ K_{\beta\alpha} & K_{\beta\beta} \end{pmatrix}^{-1} \begin{pmatrix} a \\ b \end{pmatrix} \right]. \quad (\text{C2})$$

The corresponding one-dimensional Gaussian expectation is denoted by

$$\langle gh \rangle_{K_{\alpha\alpha}} = \frac{1}{\sqrt{2\pi K_{\alpha\alpha}}} \int da g(a) h(a) \exp \left[ -\frac{a^2}{2K_{\alpha\alpha}} \right]. \quad (\text{C3})$$

The noisy Bayesian kernel is defined recursively as

$$\tilde{K}_{\alpha\beta}^{(l+1)} = C_b + C_w \langle \sigma_\alpha \sigma_\beta \rangle_{\tilde{K}^{(l)}}, \quad (\text{C4})$$

with initial condition

$$\tilde{K}_{\alpha\beta}^{(1)} = C_b + \frac{C_w}{d} \sum_{j=1}^d z_{j\alpha} z_{j\beta}. \quad (\text{C5})$$

The noisy NTK is defined recursively by

$$\tilde{\Theta}_{\alpha\beta}^{(l+1)} = C_b + C_w (\langle \sigma_\alpha \sigma_\beta \rangle_{\tilde{K}^{(l)}} + \langle \sigma'_\alpha \sigma'_\beta \rangle_{\tilde{K}^{(l)}} \tilde{\Theta}_{\alpha\beta}^{(l)}), \quad (\text{C6})$$

with initial condition

$$\tilde{\Theta}_{\alpha\beta}^{(1)} = \tilde{K}_{\alpha\beta}^{(1)}. \quad (\text{C7})$$

Since we work in the large- $d$  limit, the initial feature average obeys a central-limit expansion,

$$\frac{C_w}{d} \sum_{j=1}^d z_{j\alpha} z_{j\beta} = \frac{C_w}{d} \sum_{j=1}^d \mathbb{E}[z_{j\alpha} z_{j\beta}] + \frac{1}{\sqrt{d}} \eta_{\alpha\beta} + \mathcal{O}(d^{-1}), \quad (\text{C8})$$

where  $\eta_{\alpha\beta}$  is a mean-zero Gaussian random tensor with covariance determined by the noise model. Here and throughout this Appendix,  $\mathbb{E}[\cdot]$  denotes expectation over the input-corruption noise. Thus, the first-layer Bayesian kernel may be decomposed as

$$\tilde{K}_{\alpha\beta}^{(1)} = K_{\alpha\beta}^{(1)} + \frac{1}{\sqrt{d}}\eta_{\alpha\beta} + \mathcal{O}(d^{-1}), \quad (\text{C9})$$

where

$$K_{\alpha\beta}^{(1)} = C_b + \frac{C_w}{d} \sum_{j=1}^d \mathbb{E}[z_{j\alpha} z_{j\beta}]. \quad (\text{C10})$$

The corruption model is assumed to have a noise strength  $p \in [0, 1]$ . We define the signal strength

$$\epsilon = 1 - p, \quad (\text{C11})$$

so that  $\epsilon = 0$  is the pure-noise limit. Both  $K_{\alpha\beta}^{(1)}$  and  $\eta_{\alpha\beta}$  may depend on  $\epsilon$ . Since the expansion is around the pure-noise point, all quantities below are evaluated at  $\epsilon = 0$  unless otherwise specified. For example,

$$K_{\alpha\beta}^{(1)} = K_{\alpha\beta}^{(1)}|_{\epsilon=0}, \quad \frac{\partial K_{\alpha\beta}^{(1)}}{\partial \epsilon} = \left. \frac{\partial K_{\alpha\beta}^{(1)}(\epsilon)}{\partial \epsilon} \right|_{\epsilon=0}. \quad (\text{C12})$$

With this convention, the first-layer perturbation has the form

$$\delta K_{\alpha\beta}^{(1)} = \epsilon \frac{\partial K_{\alpha\beta}^{(1)}}{\partial \epsilon} + \frac{1}{\sqrt{d}}\eta_{\alpha\beta}. \quad (\text{C13})$$

## 2. Useful identities

We collect several identities used repeatedly below. First, Price's theorem gives the following derivatives of Gaussian expectations with respect to the covariance. For one-dimensional Gaussian expectations,

$$\frac{\partial}{\partial K_{\alpha\alpha}} \langle gh \rangle_{K_{\alpha\alpha}} = \langle g'h' \rangle_{K_{\alpha\alpha}} + \frac{1}{2} \langle g''h \rangle_{K_{\alpha\alpha}} + \frac{1}{2} \langle gh'' \rangle_{K_{\alpha\alpha}}. \quad (\text{C14})$$

For two-dimensional Gaussian expectations with  $\alpha \neq \beta$ ,

$$\frac{\partial}{\partial K_{\alpha\alpha}} \langle g_\alpha h_\beta \rangle_K = \frac{1}{2} \langle g''_\alpha h_\beta \rangle_K, \quad (\text{C15})$$

$$\frac{\partial}{\partial K_{\alpha\beta}} \langle g_\alpha h_\beta \rangle_K = \langle g'_\alpha h'_\beta \rangle_K, \quad (\text{C16})$$

$$\frac{\partial}{\partial K_{\beta\beta}} \langle g_\alpha h_\beta \rangle_K = \frac{1}{2} \langle g_\alpha h''_\beta \rangle_K. \quad (\text{C17})$$

Although these expressions are written for smooth activations, nonsmooth activations such as ReLU may be recovered by a limiting procedure.

We also use the elementary solution of the scalar recursion

$$B^{(l+1)} = a_l + b_l B^{(l)}. \quad (\text{C18})$$

It is

$$B^{(l)} = \sum_{i=1}^{l-1} a_i \prod_{j=i+1}^{l-1} b_j + B^{(1)} \prod_{i=1}^{l-1} b_i. \quad (\text{C19})$$

Next, for an  $N \times N$  matrix of the form

$$H = h_d \mathbf{I} + h_o (\mathbf{1}\mathbf{1}^T - \mathbf{I}), \quad (\text{C20})$$

where  $\mathbf{I}$  is the identity and  $\mathbf{1}$  is the vector of all ones, the inverse is

$$H^{-1} = \frac{1}{h_d - h_o} \left( \mathbf{I} - \frac{h_o}{h_d - h_o + h_o N} \mathbf{1}\mathbf{1}^T \right). \quad (\text{C21})$$

For vectors  $\mathbf{a}, \mathbf{b} \in \mathbb{R}^N$ , defining

$$\mathbb{E}_\alpha[a_\alpha] = \frac{1}{N} \sum_{\alpha=1}^N a_\alpha, \quad \mathbb{E}_\alpha[a_\alpha b_\alpha] = \frac{1}{N} \sum_{\alpha=1}^N a_\alpha b_\alpha, \quad (\text{C22})$$

we therefore have

$$\begin{aligned} \mathbf{a}^T H^{-1} \mathbf{b} &= \frac{N}{h_d - h_o} \left( \mathbb{E}_\alpha[a_\alpha b_\alpha] - \frac{h_o N}{h_d - h_o + h_o N} \mathbb{E}_\alpha[a_\alpha] \mathbb{E}_\beta[b_\beta] \right). \end{aligned} \quad (\text{C23})$$

The following layer-to-layer quantities will also be used:

$$P_{\alpha\beta}^{a \rightarrow b} = \prod_{i=a}^{b-1} \langle \sigma'_\alpha \sigma'_\beta \rangle_{K^{(i)}}, \quad (\text{C24})$$

$$U_\alpha^{a \rightarrow b} = \prod_{i=a}^{b-1} (\langle \sigma'_\alpha \sigma' \rangle_{K_{\alpha\alpha}^{(i)}} + \langle \sigma \sigma'' \rangle_{K_{\alpha\alpha}^{(i)}}), \quad (\text{C25})$$

and

$$M_{\alpha\beta}^{a \rightarrow b} = \frac{1}{2} \sum_{i=a}^{b-1} \langle \sigma''_\alpha \sigma_\beta \rangle_{K^{(i)}} U_\alpha^{a \rightarrow i} P_{\alpha\beta}^{i+1 \rightarrow b}. \quad (\text{C26})$$

The corresponding NTK-response coefficients are

$$\begin{aligned} R_\alpha^{a \rightarrow b} &= \sum_{i=a}^{b-1} (\langle \sigma'_\alpha \sigma' \rangle_{K_{\alpha\alpha}^{(i)}} + \langle \sigma \sigma'' \rangle_{K_{\alpha\alpha}^{(i)}} + [\langle \sigma'' \sigma'' \rangle_{K_{\alpha\alpha}^{(i)}} \\ &\quad + \langle \sigma' \sigma''' \rangle_{K_{\alpha\alpha}^{(i)}}] \Theta_{\alpha\alpha}^{(i)}) U_\alpha^{a \rightarrow i} P_{\alpha\alpha}^{i+1 \rightarrow b} + P_{\alpha\alpha}^{a \rightarrow b}, \end{aligned} \quad (\text{C27})$$

$$\begin{aligned} S_{\alpha\beta}^{a \rightarrow b} &= \sum_{i=a}^{b-1} \left[ \frac{1}{2} (\langle \sigma''_\alpha \sigma_\beta \rangle_{K^{(i)}} + \Theta_{\alpha\beta}^{(i)} \langle \sigma''_\alpha \sigma'_\beta \rangle_{K^{(i)}}) U_\alpha^{a \rightarrow i} \right. \\ &\quad \left. + (\langle \sigma'_\alpha \sigma'_\beta \rangle_{K^{(i)}} + \Theta_{\alpha\beta}^{(i)} \langle \sigma''_\alpha \sigma''_\beta \rangle_{K^{(i)}}) M_{\alpha\beta}^{a \rightarrow i} \right] P_{\alpha\beta}^{i+1 \rightarrow b}, \end{aligned} \quad (\text{C28})$$

$$\begin{aligned} T_{\alpha\beta}^{a \rightarrow b} &= \sum_{i=a}^{b-1} (\langle \sigma'_\alpha \sigma'_\beta \rangle_{K^{(i)}} + \Theta_{\alpha\beta}^{(i)} \langle \sigma''_\alpha \sigma''_\beta \rangle_{K^{(i)}}) P_{\alpha\beta}^{a \rightarrow i} P_{\alpha\beta}^{i+1 \rightarrow b} \\ &\quad + P_{\alpha\beta}^{a \rightarrow b}. \end{aligned} \quad (\text{C29})$$

All quantities in Eqs. (C24)–(C29) are evaluated in the pure-noise limit. Therefore, their values depend only on whether the indices are training-training, test-training, or test-test indices; they do not otherwise depend on the individual data points.

## 3. Influence of input perturbations on the output

We now compute how a perturbation in the first-layer Bayesian kernel propagates to the output. Let

$$\begin{aligned} \tilde{K}_{\alpha\beta}^{(l)} &= K_{\alpha\beta}^{(l)} + \delta K_{\alpha\beta}^{(l)} + \mathcal{O}(\delta K^2), \\ \tilde{\Theta}_{\alpha\beta}^{(l)} &= \Theta_{\alpha\beta}^{(l)} + \delta \Theta_{\alpha\beta}^{(l)} + \mathcal{O}(\delta K^2). \end{aligned} \quad (\text{C30})$$

The unperturbed kernels satisfy the pure-noise recursions

$$\Theta_{\alpha\beta}^{(l+1)} = C_b + C_w (\langle \sigma_\alpha \sigma_\beta \rangle_{K^{(l)}} + \langle \sigma'_\alpha \sigma'_\beta \rangle_{K^{(l)}} \Theta_{\alpha\beta}^{(l)}), \quad (C31)$$

$$K_{\alpha\beta}^{(l+1)} = C_b + C_w \langle \sigma_\alpha \sigma_\beta \rangle_{K^{(l)}}. \quad (C32)$$

The linear perturbations obey

$$\begin{aligned} \frac{1}{C_w} \delta K_{\alpha\beta}^{(l+1)} = & \delta_{\alpha\beta} (\langle \sigma' \sigma' \rangle_{K_{\alpha\alpha}^{(l)}} + \langle \sigma \sigma'' \rangle_{K_{\alpha\alpha}^{(l)}}) \delta K_{\alpha\alpha}^{(l)} \\ & + (1 - \delta_{\alpha\beta}) \left[ \frac{1}{2} \langle \sigma'' \sigma \beta \rangle_{K^{(l)}} \delta K_{\alpha\alpha}^{(l)} \right. \\ & \left. + \langle \sigma'_\alpha \sigma'_\beta \rangle_{K^{(l)}} \delta K_{\alpha\beta}^{(l)} + \frac{1}{2} \langle \sigma_\alpha \sigma''_\beta \rangle_{K^{(l)}} \delta K_{\beta\beta}^{(l)} \right], \end{aligned} \quad (C33)$$

$$\begin{aligned} \frac{1}{C_w} \delta \Theta_{\alpha\beta}^{(l+1)} = & \delta_{\alpha\beta} \left[ (\langle \sigma' \sigma' \rangle_{K_{\alpha\alpha}^{(l)}} + \langle \sigma \sigma'' \rangle_{K_{\alpha\alpha}^{(l)}} + [\langle \sigma'' \sigma'' \rangle_{K_{\alpha\alpha}^{(l)}} \right. \\ & \left. + \langle \sigma' \sigma''' \rangle_{K_{\alpha\alpha}^{(l)}}] \Theta_{\alpha\alpha}^{(l)}) \delta K_{\alpha\alpha}^{(l)} + \langle \sigma' \sigma' \rangle_{K_{\alpha\alpha}^{(l)}} \delta \Theta_{\alpha\alpha}^{(l)} \right] \\ & + (1 - \delta_{\alpha\beta}) \left[ \frac{1}{2} (\langle \sigma'' \sigma \beta \rangle_{K^{(l)}} + \langle \sigma''' \sigma'_\beta \rangle_{K^{(l)}} \Theta_{\alpha\beta}^{(l)}) \right. \\ & \times \delta K_{\alpha\alpha}^{(l)} + (\langle \sigma'_\alpha \sigma'_\beta \rangle_{K^{(l)}} + \langle \sigma''_\alpha \sigma''_\beta \rangle_{K^{(l)}} \Theta_{\alpha\beta}^{(l)}) \delta K_{\alpha\beta}^{(l)} \\ & + \frac{1}{2} (\langle \sigma_\alpha \sigma''_\beta \rangle_{K^{(l)}} + \langle \sigma'_\alpha \sigma'''_\beta \rangle_{K^{(l)}} \Theta_{\alpha\beta}^{(l)}) \delta K_{\beta\beta}^{(l)} \\ & \left. + \langle \sigma'_\alpha \sigma'_\beta \rangle_{K^{(l)}} \delta \Theta_{\alpha\beta}^{(l)} \right]. \end{aligned} \quad (C34)$$

Given the solution of the pure-noise Bayesian-kernel recursion, the pure-noise NTK has the closed-form solution

$$\frac{1}{C_w^{l-1}} \Theta_{\alpha\beta}^{(l)} = \sum_{i=1}^{l-1} (C_b + C_w \langle \sigma_\alpha \sigma_\beta \rangle_{K^{(i)}}) C_w^{-i} P_{\alpha\beta}^{i+1 \rightarrow l} + P_{\alpha\beta}^{1 \rightarrow l} \Theta_{\alpha\beta}^{(1)}, \quad (C35)$$

and the perturbations propagate as

$$\begin{aligned} \frac{1}{C_w^{l-1}} \delta K_{\alpha\beta}^{(l)} = & \delta_{\alpha\beta} U_\alpha^{1 \rightarrow l} \delta K_{\alpha\alpha}^{(1)} + (1 - \delta_{\alpha\beta}) \\ & \times (M_{\alpha\beta}^{1 \rightarrow l} \delta K_{\alpha\alpha}^{(1)} + M_{\beta\alpha}^{1 \rightarrow l} \delta K_{\beta\beta}^{(1)} + P_{\alpha\beta}^{1 \rightarrow l} \delta K_{\alpha\beta}^{(1)}), \end{aligned} \quad (C36)$$

$$\begin{aligned} \frac{1}{C_w^{L-1}} \delta \Theta_{\alpha\beta}^{(L)} = & \delta_{\alpha\beta} R_\alpha^{1 \rightarrow L} \delta K_{\alpha\alpha}^{(1)} + (1 - \delta_{\alpha\beta}) \\ & \times (S_{\alpha\beta}^{1 \rightarrow L} \delta K_{\alpha\alpha}^{(1)} + S_{\beta\alpha}^{1 \rightarrow L} \delta K_{\beta\beta}^{(1)} + T_{\alpha\beta}^{1 \rightarrow L} \delta K_{\alpha\beta}^{(1)}). \end{aligned} \quad (C37)$$

At pure noise, the training-training NTK at layer  $L$  has the permutation-symmetric form

$$\frac{1}{C_w^{L-1}} \Theta_{\chi\chi}^{(L)} = \theta_d \mathbf{I} + \theta_o (\mathbf{1}\mathbf{1}^T - \mathbf{I}), \quad (C38)$$

while the test-training NTK has the form

$$\frac{1}{C_w^{L-1}} \Theta_{*\chi}^{(L)} = \theta_* \mathbf{1}^T. \quad (C39)$$

Here,  $\theta_d$ ,  $\theta_o$ , and  $\theta_*$  are constants determined by the architecture and the pure-noise kernel. We define

$$\psi = \frac{N}{\theta_d - \theta_o + \theta_o N},$$

$$\text{Cov}_\alpha^1(a_\alpha, b_\alpha) = \mathbb{E}_\alpha[a_\alpha b_\alpha] - \theta_o \psi \mathbb{E}_\alpha[a_\alpha] \mathbb{E}_\alpha[b_\alpha]. \quad (C40)$$

Expanding the kernel-regression output to first order in  $\delta\Theta$  gives

$$\begin{aligned} f_i(x_*) = & [\Theta_{*\chi}^{(L)} [\Theta_{\chi\chi}^{(L)}]^{-1} + \delta \Theta_{*\chi}^{(L)} [\Theta_{\chi\chi}^{(L)}]^{-1} \\ & - \Theta_{*\chi}^{(L)} [\Theta_{\chi\chi}^{(L)}]^{-1} \delta \Theta_{\chi\chi}^{(L)} [\Theta_{\chi\chi}^{(L)}]^{-1}] y_i. \end{aligned} \quad (C41)$$

Substituting the pure-noise inverse and the perturbative solution gives

$$\begin{aligned} f_i(x_*) = & \theta_* \psi \mathbb{E}_\alpha[y_{\alpha i}] \\ & + \frac{N}{\theta_d - \theta_o} \left( (S_{\chi*}^{1 \rightarrow L} - \theta_* \psi S_{\chi\chi}^{1 \rightarrow L}) \text{Cov}_\alpha^1(\delta K_{\alpha\alpha}^{(1)}, y_{\alpha i}) \right. \\ & + T_{*\chi}^{1 \rightarrow L} \text{Cov}_\alpha^1(\delta K_{*\alpha}^{(1)}, y_{\alpha i}) \\ & - \theta_* \psi \left[ T_{\chi\chi}^{1 \rightarrow L} \mathbb{E}_\alpha[\text{Cov}_\beta^1(\delta K_{\alpha\beta}^{(1)}, y_{\beta i})] \right. \\ & \left. \left. + S_{\chi\chi}^{1 \rightarrow L} (1 - \theta_o \psi) \mathbb{E}_\alpha[\delta K_{\alpha\alpha}^{(1)}] \mathbb{E}_\alpha[y_{\alpha i}] \right] \right) \\ & + \mathcal{O}(\delta K N^0) + \mathcal{O}(\delta K^2). \end{aligned} \quad (C42)$$

#### 4. Output fluctuations under Gaussian kernel perturbations

We now isolate the fluctuation term obtained by taking

$$\delta K_{\alpha\beta}^{(1)} = \frac{1}{\sqrt{d}} \eta_{\alpha\beta}, \quad (C43)$$

where  $\eta_{\alpha\beta}$  is a mean-zero Gaussian random tensor. Since Eq. (C42) is linear in  $\eta_{\alpha\beta}$ , the induced output fluctuation is also Gaussian. It is therefore enough to compute its covariance.

The basic identity is

$$\begin{aligned} \mathbb{E}[\text{Cov}_\beta^1(\eta_{\alpha\beta}, y_{\beta i}) \text{Cov}_\delta^1(\eta_{\gamma\delta}, y_{\delta j})] \\ = \frac{1}{N} \mathbb{Cov}(\eta_{\alpha\beta}, \eta_{\beta\gamma}) \text{Cov}_\alpha^2(y_{\beta i}, y_{\gamma j}), \end{aligned} \quad (C44)$$

where

$$\text{Cov}_\alpha^2(a_\alpha, b_\alpha) = \mathbb{E}_\alpha[a_\alpha b_\alpha] - (2\theta_o \psi - (\theta_o \psi)^2) \mathbb{E}_\alpha[a_\alpha] \mathbb{E}_\alpha[b_\alpha]. \quad (C45)$$

This identity follows from the i.i.d. structure of the input noise: A covariance between two  $\eta$  variables vanishes unless the two index pairs share at least one data index.

For large  $N$ , the required contractions are

$$\begin{aligned} \mathbb{E}[\text{Cov}_\alpha^1(\eta_{\alpha\alpha}, y_{\alpha i}) \text{Cov}_\beta^1(\eta_{\beta\beta}, y_{\beta j})] \\ = \frac{1}{N} \mathbb{V}\text{ar}(\eta_{\alpha\alpha}) \text{Cov}_\beta^2(y_{\beta i}, y_{\beta j}), \end{aligned} \quad (C46)$$

$$\begin{aligned} \mathbb{E}[\text{Cov}_\alpha^1(\eta_{\alpha\alpha}, y_{\alpha i}) \text{Cov}_\beta^1(\eta_{*\beta}, y_{\beta j})] \\ = \frac{1}{N} \mathbb{Cov}(\eta_{\alpha\alpha}, \eta_{*\alpha}) \text{Cov}_\beta^2(y_{\beta i}, y_{\beta j}), \end{aligned} \quad (C47)$$

$$\begin{aligned} & \mathbb{E}[\text{Cov}_\alpha^1(\eta_{\alpha\alpha}, y_{\alpha i}) \text{E}_\beta[\text{Cov}_\gamma^1(\eta_{\beta\gamma}, y_{\gamma j})]] \\ &= \frac{1}{N} \text{Cov}(\eta_{\alpha\alpha}, \eta_{\alpha\beta}) (\text{Cov}_\gamma^2(y_{\gamma i}, y_{\gamma j}) \\ &+ (1 - \theta_o \psi)^2 \text{E}_\gamma[y_{\gamma i}] \text{E}_\delta[y_{\delta j}]) + \mathcal{O}\left(\frac{1}{N^2}\right), \end{aligned} \quad (\text{C48})$$

$$\begin{aligned} & \mathbb{E}[\text{Cov}_\alpha^1(\eta_{\alpha\alpha}, y_{\alpha i}) \text{E}_\beta[\eta_{\beta\beta}] \text{E}_\gamma[y_{\gamma j}]] \\ &= \frac{1}{N} \mathbb{V}\text{ar}(\eta_{\alpha\alpha}) (1 - \theta_o \psi) \text{E}_\beta[y_{\beta i}] \text{E}_\gamma[y_{\gamma j}], \end{aligned} \quad (\text{C49})$$

$$\begin{aligned} & \mathbb{E}[\text{Cov}_\alpha^1(\eta_{*\alpha}, y_{\alpha i}) \text{Cov}_\beta^1(\eta_{*\beta}, y_{\beta j})] \\ &= \frac{1}{N} \mathbb{V}\text{ar}(\eta_{*\alpha}) \text{Cov}_\beta^2(y_{\beta i}, y_{\beta j}), \end{aligned} \quad (\text{C50})$$

$$\begin{aligned} & \mathbb{E}[\text{Cov}_\alpha^1(\eta_{*\alpha}, y_{\alpha i}) \text{E}_\beta[\text{Cov}_\gamma^1(\eta_{\beta\gamma}, y_{\gamma j})]] \\ &= \frac{1}{N} \text{Cov}(\eta_{*\alpha}, \eta_{\alpha\beta}) (\text{Cov}_\gamma^2(y_{\gamma i}, y_{\gamma j}) \\ &+ (1 - \theta_o \psi)^2 \text{E}_\gamma[y_{\gamma i}] \text{E}_\delta[y_{\delta j}]) + \mathcal{O}\left(\frac{1}{N^2}\right), \end{aligned} \quad (\text{C51})$$

$$\begin{aligned} & \mathbb{E}[\text{Cov}_\alpha^1(\eta_{*\alpha}, y_{\alpha i}) \text{E}_\beta[\eta_{\beta\beta}] \text{E}_\gamma[y_{\gamma j}]] \\ &= \frac{1}{N} \text{Cov}(\eta_{*\alpha}, \eta_{\alpha\alpha}) (1 - \theta_o \psi) \text{E}_\beta[y_{\beta i}] \text{E}_\gamma[y_{\gamma j}], \end{aligned} \quad (\text{C52})$$

$$\begin{aligned} & \mathbb{E}[\text{E}_\alpha[\text{Cov}_\beta^1(\eta_{\alpha\beta}, y_{\beta i})] \text{E}_\gamma[\text{Cov}_\delta^1(\eta_{\gamma\delta}, y_{\delta j})]] \\ &= \frac{1}{N} \text{Cov}(\eta_{\alpha\beta}, \eta_{\beta\gamma}) (\text{Cov}_\delta^2(y_{\delta i}, y_{\delta j}) \\ &+ 3(1 - \theta_o \psi)^2 \text{E}_\delta[y_{\delta i}] \text{E}_\lambda[y_{\lambda j}]) + \mathcal{O}\left(\frac{1}{N^2}\right), \end{aligned} \quad (\text{C53})$$

$$\begin{aligned} & \mathbb{E}[\text{E}_\alpha[\text{Cov}_\beta^1(\eta_{\alpha\beta}, y_{\beta i})] \text{E}_\gamma[\eta_{\gamma\gamma}] \text{E}_\delta[y_{\delta j}]] \\ &= \frac{2}{N} \text{Cov}(\eta_{\alpha\beta}, \eta_{\beta\gamma}) (1 - \theta_o \psi) \text{E}_\gamma[y_{\gamma i}] \text{E}_\delta[y_{\delta j}] \\ &+ \mathcal{O}\left(\frac{1}{N^2}\right), \end{aligned} \quad (\text{C54})$$

$$\begin{aligned} & \mathbb{E}[\text{E}_\alpha[\eta_{\alpha\alpha}] \text{E}_\beta[y_{\beta i}] \text{E}_\gamma[\eta_{\gamma\gamma}] \text{E}_\delta[y_{\delta j}]] \\ &= \frac{1}{N} \mathbb{V}\text{ar}(\eta_{\alpha\alpha}) \text{E}_\beta[y_{\beta i}] \text{E}_\gamma[y_{\gamma j}]. \end{aligned} \quad (\text{C55})$$

The variances and covariances appearing here are scalars evaluated at pure noise. In particular, they have no explicit dependence on the clean training data, because the clean signal has been set to zero before evaluating the fluctuation covariance.

Combining these contractions gives

$$\begin{aligned} & \text{Cov}(f_i(x_*), f_j(x_*)) \\ &= \frac{N/d}{(\theta_d - \theta_o)^2} \tilde{V}_{ij} + \mathcal{O}\left(\frac{1}{d^2}\right) + \mathcal{O}\left(\frac{N^0}{d}\right) + \mathcal{O}\left(\frac{N}{d^2}\right), \end{aligned} \quad (\text{C56})$$

where  $\tilde{V}_{ij}$  is an architecture- and noise-dependent coefficient matrix. For  $C$  equally partitioned classes,

$$\text{Cov}_\alpha^2(y_{\alpha i}, y_{\alpha j}) = \frac{1}{C} \left( \delta_{ij} - \frac{1}{C} [2\theta_o \psi - (\theta_o \psi)^2] \right). \quad (\text{C57})$$

Thus, the coefficient matrix can be written as

$$\tilde{V}_{ij} = \delta_{ij} \tilde{V}_d + (1 - \delta_{ij}) \tilde{V}_o, \quad (\text{C58})$$

for scalars  $\tilde{V}_d$  and  $\tilde{V}_o$ . Since the  $\text{Cov}^2$  has an overall factor of  $1/C$ , we can pull this out by defining  $V_{ij} = C\tilde{V}_{ij}$ . Thus, the output fluctuations may be written as

$$f_i(x_*) - \mathbb{E}[f_i(x_*)] = \sqrt{\frac{N/C}{d}} \frac{\sqrt{V_d}}{\theta_d - \theta_o} \hat{\xi}_i, \quad \hat{\xi} \sim \mathcal{N}(0, \Sigma), \quad (\text{C59})$$

where

$$\Sigma_{ij} = \delta_{ij} + (1 - \delta_{ij}) \frac{V_o}{V_d}. \quad (\text{C60})$$

### 5. The output for an arbitrary i.i.d. noise model

For an arbitrary i.i.d. noise model, the first-layer perturbation is

$$\delta K_{\alpha\beta}^{(1)} = \epsilon \frac{\partial K_{\alpha\beta}^{(1)}}{\partial \epsilon} + \frac{1}{\sqrt{d}} \eta_{\alpha\beta}. \quad (\text{C61})$$

Substituting this into Eq. (C42), and using the fluctuation result Eq. (C59), gives the general high-noise output. Here, we assume the dataset has been partitioned into  $C$  equally represented classes. We define

$$\Delta E_\alpha^i[a_\alpha] = \frac{C}{N} \sum_{\alpha \in i} a_\alpha - \theta_o \psi \frac{1}{N} \sum_{\alpha=1}^N a_\alpha, \quad (\text{C62})$$

where  $\alpha \in i$  means that the sum is restricted to training examples in class  $i$ . Since  $\text{E}_\alpha[y_{\alpha i}] = 1/C$ , we have

$$\text{Cov}_\alpha^1(a_\alpha, y_{\alpha i}) = \frac{1}{C} \Delta E_\alpha^i[a_\alpha]. \quad (\text{C63})$$

The output therefore becomes

$$\begin{aligned} f_i(x_*; \epsilon) &= \frac{\theta_* \psi}{C} + \sqrt{\frac{N/C}{d}} \frac{\sqrt{V_d}}{\theta_d - \theta_o} \hat{\xi}_i \\ &+ \frac{\epsilon N/C}{\theta_d - \theta_o} \left( (S_{\chi^*}^{1 \rightarrow L} - \theta_* \psi S_{\chi\chi}^{1 \rightarrow L}) \Delta E_\alpha^i \left[ \frac{\partial K_{\alpha\alpha}^{(1)}}{\partial \epsilon} \right] \right. \\ &+ T_{*\chi}^{1 \rightarrow L} \Delta E_\alpha^i \left[ \frac{\partial K_{*\alpha}^{(1)}}{\partial \epsilon} \right] \\ &- \theta_* \psi \left( T_{\chi\chi}^{1 \rightarrow L} \text{E}_\alpha \left[ \Delta E_\beta^i \left[ \frac{\partial K_{\alpha\beta}^{(1)}}{\partial \epsilon} \right] \right] \right. \\ &+ S_{\chi\chi}^{1 \rightarrow L} (1 - \theta_o \psi) \text{E}_\alpha \left[ \frac{\partial K_{\alpha\alpha}^{(1)}}{\partial \epsilon} \right] \left. \right) \\ &+ \mathcal{O}(\epsilon N^0) + \mathcal{O}(\epsilon^2) \\ &+ \mathcal{O}(N^0/\sqrt{d}) + \mathcal{O}(\epsilon N/\sqrt{d}) + \mathcal{O}(N/d). \end{aligned} \quad (\text{C64})$$

We now impose the normalization used in the main text,

$$\frac{1}{d} \sum_{j=1}^d x_{j\alpha} = 0, \quad \frac{1}{d} \sum_{j=1}^d x_{j\alpha}^2 = 1, \quad (\text{C65})$$

for both training and test data.

Here, we consider two general noise models. Additive noise with any mean-zero, unit-variance noise variable,

$$\tilde{x}_{j\chi} = \epsilon x_{j\chi} + (1 - \epsilon) \xi_{j\chi}, \quad \mathbb{E}[\xi_{j\chi}] = 0, \quad \mathbb{E}[\xi_{j\chi}^2] = 1, \quad (\text{C66})$$

for which the first-layer derivatives are

$$\frac{\partial K_{* \chi}^{(1)}}{\partial \epsilon} = \frac{C_w}{d} \sum_{j=1}^d x_{j*} x_{j\chi}, \quad \frac{\partial K_{\chi \chi'}^{(1)}}{\partial \epsilon} = -2C_w \delta_{\chi \chi'}, \quad (\text{C67})$$

and replacement noise with an arbitrary replacement distribution  $u$ ,

$$\tilde{x}_{j\chi} = (1 - b_{j\chi}) x_{j\chi} + b_{j\chi} u_{j\chi}, \quad \text{Pr}(b_{j\chi} = 0) = \epsilon, \quad (\text{C68})$$

for which the normalized data imply

$$\frac{\partial K_{* \chi}^{(1)}}{\partial \epsilon} = \frac{C_w}{d} \sum_{j=1}^d x_{j*} x_{j\chi}, \quad (\text{C69})$$

$$\frac{\partial K_{\chi \chi'}^{(1)}}{\partial \epsilon} = C_w \delta_{\chi \chi'} (1 - \mathbb{E}[u^2]) - 2C_w (1 - \delta_{\chi \chi'}) \mathbb{E}[u]^2. \quad (\text{C70})$$

Thus, for both additive noise and replacement noise, the only class-dependent data dependence in Eq. (C64) comes from

$$\Delta E_{\alpha}^i \left[ \frac{\partial K_{* \alpha}^{(1)}}{\partial \epsilon} \right] = C_w \Delta E_{\alpha}^i \left[ \frac{1}{d} \sum_{j=1}^d x_{j*} x_{j\alpha} \right]. \quad (\text{C71})$$

All other quantities in Eq. (C64), including  $\theta_*$ ,  $\theta_d$ ,  $\theta_o$ ,  $S$ ,  $T$ , and  $V_d$ , are constants determined only by the architecture and the noise distribution. This is because, under the normalization of Eq. (C65), the pure-noise kernels depend on the data only through fixed quantities such as  $\frac{1}{d} \sum_j x_{j\alpha}$  and  $\frac{1}{d} \sum_j x_{j\alpha}^2$ .

Therefore, for either noise model,

$$\begin{aligned} f_i(x_*; \epsilon) = & \text{const} + A^{(L)} \sqrt{\frac{N/C}{d}} \hat{\xi}_i \\ & + B^{(L)} \epsilon \frac{N}{C} \Delta E_{\alpha}^i \left[ \frac{1}{d} \sum_{j=1}^d x_{j*} x_{j\alpha} \right] \\ & + \mathcal{O}(\epsilon N^0) + \mathcal{O}(\epsilon^2) + \mathcal{O}(N^0/\sqrt{d}) \\ & + \mathcal{O}(\epsilon N/\sqrt{d}) + \mathcal{O}(N/d), \end{aligned} \quad (\text{C72})$$

where  $A^{(L)}$  and  $B^{(L)}$  are architecture- and noise-distribution-dependent constants.

Finally, to obtain the expression quoted in the main text, note that when  $\theta_o \neq 0$ ,

$$\theta_o \psi = \frac{\theta_o N}{\theta_d - \theta_o + \theta_o N} = 1 + \mathcal{O}(N^{-1}). \quad (\text{C73})$$

The exceptional case  $\theta_o = 0$  can occur in special symmetry-preserving settings, such as when the activation is chosen to be odd and  $C_b = 0$ . This is a nongeneric degeneracy rather than a fundamental obstruction. Indeed, adding any positive bias variance  $C_b > 0$ , however small, breaks this exact odd symmetry at the level of the kernel recursion and generically produces a nonzero off-diagonal pure-noise kernel  $\theta_o$ . Thus, the case  $\theta_o = 0$  can be avoided by an arbitrarily small and

standard modification of the initialization. In the generic case  $\theta_o \neq 0$ , we therefore have

$$\Delta E_{\alpha}^i[a_{\alpha}] = \frac{C}{N} \sum_{\alpha \in i} a_{\alpha} - \frac{1}{N} \sum_{\alpha=1}^N a_{\alpha} + \mathcal{O}(N^{-1}). \quad (\text{C74})$$

Defining the empirical class mean (centroid) and global mean by

$$\bar{x}_j^i = \frac{C}{N} \sum_{\alpha \in i} x_{j\alpha}, \quad \bar{x}_j = \frac{1}{N} \sum_{\alpha=1}^N x_{j\alpha}, \quad (\text{C75})$$

we obtain Eq. (13),

$$\begin{aligned} f_i(x_*; \epsilon) = & \text{const} + A^{(L)} \sqrt{\frac{N/C}{d}} \hat{\xi}_i \\ & + B^{(L)} \epsilon \frac{N}{C} \frac{1}{d} \sum_{j=1}^d x_{j*} (\bar{x}_j^i - \bar{x}_j) \\ & + \mathcal{O}(\epsilon N^0) + \mathcal{O}(\epsilon^2) + \mathcal{O}(N^0/\sqrt{d}) \\ & + \mathcal{O}(\epsilon N/\sqrt{d}) + \mathcal{O}(N/d). \end{aligned} \quad (\text{C76})$$

Using  $\epsilon = 1 - p$  gives the high-noise expression stated in the main text. The leading class-dependent term is therefore the overlap between the clean test point and the centered empirical class centroid. Consequently, after averaging over the finite- $d$  fluctuations, the network implements a nearest-class-mean rule in the high-noise limit.

#### APPENDIX D: NON-NEGATIVITY OF THE COEFFICIENT MULTIPLYING THE CENTROID

For the centroid term in Eq. (13) to have the correct orientation, its coefficient  $B^{(L)}$  must be non-negative. We show here that this is guaranteed by the normalization convention and the unit-variance noise models adopted in the main text.

According to Eq. (C64), the sign of  $B^{(L)}$  is that of  $T_{* \chi}^{(L)}$ , so it is enough to establish  $T_{* \chi}^{(L)} \geq 0$ . The recursion that generates  $T_{* \chi}^{(L)}$  (Appendix C) involves only the initialization constants  $C_b \geq 0$  and  $C_w > 0$ , the test-train kernels  $K_{* \chi}^{(\ell)}$  and NTKs  $\Theta_{* \chi}^{(\ell)}$ , and the two-dimensional Gaussian expectations

$$\langle h_* h_{\chi} \rangle_{K^{(\ell)}}, \quad (\text{D1})$$

in which  $h$  stands for the activation or one of its derivatives, evaluated on a pair of jointly Gaussian preactivations. The non-negativity of  $T_{* \chi}^{(L)}$  follows once each of these ingredients is shown to be non-negative at every layer.

The essential observation concerns the Gaussian expectation itself. Let  $(z_*, z_{\chi})$  be jointly Gaussian with zero mean and covariance

$$K = \begin{pmatrix} K_{**} & K_{* \chi} \\ K_{\chi *} & K_{\chi \chi} \end{pmatrix}, \quad (\text{D2})$$

and suppose the two preactivations share a common variance and a non-negative correlation,

$$K_{**} = K_{\chi \chi} > 0, \quad K_{* \chi} \geq 0. \quad (\text{D3})$$

Because the two marginals have the common variance  $K_{**}$ , we may rescale to standard normal variables, writing  $z_* =$

$\sqrt{K_{**}}X$  and  $z_\chi = \sqrt{K_{**}}Y$  with  $(X, Y)$  jointly standard normal. Their correlation is inherited from the off-diagonal entry of  $K$ ,

$$\rho = \frac{K_{*\chi}}{K_{**}} \in [0, 1], \quad (\text{D4})$$

where  $\rho \geq 0$  follows from  $K_{*\chi} \geq 0$ , and  $\rho \leq 1$  because  $K$  is a valid covariance matrix. We now expand the activation in the Hermite polynomials  $H_n$ , orthogonal with respect to the standard Gaussian weight ( $H_0 = 1, H_1 = x, H_2 = x^2 - 1$ , and so on) [40,41],

$$h(\sqrt{K_{**}}x) = \sum_{n=0}^{\infty} c_n H_n(x). \quad (\text{D5})$$

Using the bivariate identity  $\mathbb{E}[H_m(X)H_n(Y)] = \delta_{mn} n! \rho^n$ , the expectation collapses to a single sum over the diagonal,

$$\langle h_* h_\chi \rangle_K = \mathbb{E}[h(z_*)h(z_\chi)] = \sum_{n=0}^{\infty} c_n^2 n! \rho^n \geq 0, \quad (\text{D6})$$

every term of which is non-negative because  $\rho \geq 0$ . A common variance together with a non-negative correlation is therefore sufficient to render  $\langle h_* h_\chi \rangle_K$  non-negative for any activation  $h$  square integrable with respect to a Gaussian measure; in particular, this holds for  $h = \sigma, \sigma', \sigma''$ .

It remains to verify that the conditions of Eq. (D3) are met at every layer. At the first layer, the kernel is fixed directly by the input overlaps,

$$K^{(1)} = C_b + C_w \begin{pmatrix} \frac{1}{d} \sum_j x_{j*}^2 & \mathbb{E}[u] \frac{1}{d} \sum_j x_{j*} \\ \mathbb{E}[u] \frac{1}{d} \sum_j x_{j*} & \mathbb{E}[u^2] \end{pmatrix}, \quad (\text{D7})$$

the constant  $C_b$  being added to every entry. The normalization

$$\frac{1}{d} \sum_j x_{j*} = 0, \quad \frac{1}{d} \sum_j x_{j*}^2 = 1 \quad (\text{D8})$$

annihilates the off-diagonal signal and fixes the test-point variance to unity, leaving  $K_{**}^{(1)} = C_b \geq 0$ . The diagonal entries,  $C_b + C_w$  and  $C_b + C_w \mathbb{E}[u^2]$ , coincide precisely when  $\mathbb{E}[u^2] = 1$ —which is the case for both noise models employed here, since replacement noise with  $u \sim \text{Unif}[-\sqrt{3}, \sqrt{3}]$  has  $\mathbb{E}[u] = 0$  and  $\mathbb{E}[u^2] = 1$ , and the additive-Gaussian noise is taken with zero mean and unit variance. The layer-1 kernel thus satisfies Eq. (D3), with  $K_{**}^{(1)} > 0$  assured by  $C_w > 0$ .

This structure is inherited under the layer-to-layer recursion. The diagonal entries evolve according to  $K_{\alpha\alpha}^{(\ell+1)} = C_b + C_w \langle \sigma^2 \rangle_{K_{\alpha\alpha}^{(\ell)}}$ , which depends on the layer- $\ell$  kernel only through the single diagonal entry  $K_{\alpha\alpha}^{(\ell)}$ ; the two diagonal entries are therefore governed by the same map, so equal diagonals at layer  $\ell$  yield equal diagonals at layer  $\ell + 1$ . The off-diagonal entry evolves according to

$$K_{*\chi}^{(\ell+1)} = C_b + C_w \langle \sigma_* \sigma_\chi \rangle_{K^{(\ell)}} \geq C_b \geq 0, \quad (\text{D9})$$

where the inequality holds because the quantity  $\langle \sigma_* \sigma_\chi \rangle_{K^{(\ell)}}$  is itself non-negative: This is precisely Eq. (D6) with  $h = \sigma$ , applied to the layer- $\ell$  kernel, which by hypothesis has equal diagonals and a non-negative off-diagonal entry. Both conditions of Eq. (D3) are thus passed from layer  $\ell$  to layer  $\ell + 1$ , and since they hold at the first layer they hold at every layer

up to  $L$ . The same reasoning applied to the NTK recursion  $\Theta_{*\chi}^{(\ell+1)} = K_{*\chi}^{(\ell+1)} + C_w \langle \sigma'_* \sigma'_\chi \rangle_{K^{(\ell)}} \Theta_{*\chi}^{(\ell)}$ , beginning from  $\Theta_{*\chi}^{(1)} = K_{*\chi}^{(1)} = C_b \geq 0$ , shows that  $\Theta_{*\chi}^{(\ell)} \geq 0$  throughout.

With the common-variance and non-negative-correlation properties secured at every layer, the Gaussian expectations  $\langle h_* h_\chi \rangle_{K^{(\ell)}}$ —for  $h = \sigma, \sigma', \sigma''$  alike—are all non-negative, as are  $K_{*\chi}^{(\ell)}$  and  $\Theta_{*\chi}^{(\ell)}$ . Every term of the recursion for  $T_{*\chi}^{(L)}$  is then a sum of products of non-negative quantities, so that

$$T_{*\chi}^{(L)} \geq 0, \quad \text{and hence,} \quad B^{(L)} \geq 0. \quad (\text{D10})$$

The centroid term in Eq. (13) therefore carries a non-negative coefficient under our normalization convention and unit-variance noise models.

## APPENDIX E: EMPIRICAL VALIDATION OF THE CENTROID MODEL FOR ADDITIVE-GAUSSIAN NOISE

We now show that the same high-noise picture extends to additive-Gaussian corruption. In the insets of Fig. 7, we compare the ensemble-averaged MLP logits on the full MNIST, Fashion-MNIST, and KMNIST test sets with the prediction obtained from the mean of Eq. (16) for noise strength  $p = 0.97$ . Using a global fit for the coefficients  $a$  and  $b$ , we observe excellent agreement. This indicates that, once averaged over noise realizations, the network outputs are well described by the mean centroid-based predictor in the additive-Gaussian setting as well. The main panels of Fig. 7 compare the clean test accuracy of the empirical ensemble against that of the full effective model in Eq. (16), including the fluctuation term set by the fitted coefficient  $c$ . The close match shows that the centroid description accounts not only for the average logits, but also for the noise-induced variations that determine classification performance. In particular, it reproduces the decline in clean test accuracy as  $p$  increases. The figure also shows where the empirical networks begin to deviate from the leading-order centroid prediction, indicating the growing importance of higher-order terms in  $(1 - p)$ . In our experiments, these deviations become visible for  $p \lesssim 0.9$ .

We next examine the  $N, d$  dependence of the empirical SNR and test accuracy under additive-Gaussian noise, using MNIST at fixed  $p = 0.97$  and  $C = 10$ . Figure 8(a) compares the mean  $\ln \text{SNR}$  with  $\ln(Nd)$ , using the same estimator and averaging procedure as in the main text. This provides a direct test of the  $\text{SNR} \propto Nd$  dependence suggested by Eq. (17). Panel (b) shows that clean test accuracy increases systematically with both  $N$  and  $d$ , consistent with the proposed scaling.

## APPENDIX F: TEST ACCURACY OVER THE FULL NOISE RANGE

Figure 9 extends the centroid-model comparison to the full range  $p \in [0, 1]$ . The replacement-noise and additive-Gaussian experiments use the same hyperparameters and evaluation procedures as Figs. 4 and 7, respectively. The extended curves show how the agreement in the high-noise regime gives way to a growing accuracy advantage for the neural networks as  $p$  decreases. This comparison illustrates the limits of the leading-order centroid description and the additional classification performance attained by the networks outside that regime.

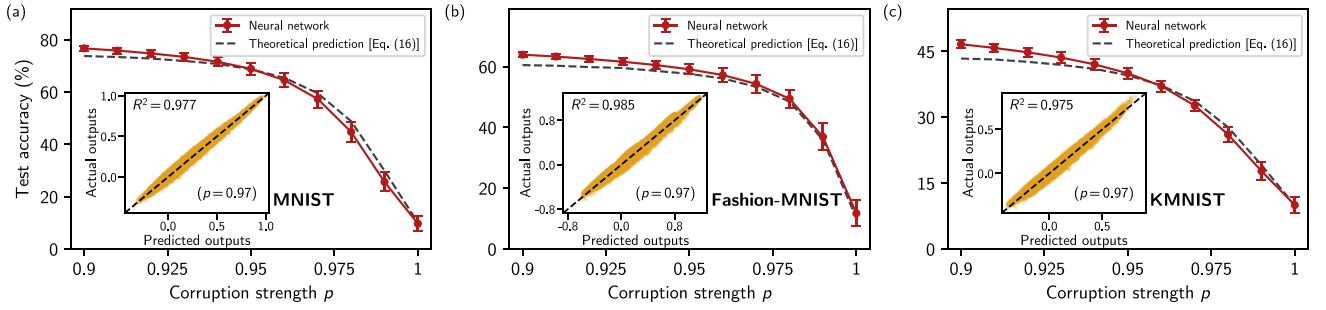


FIG. 7. Numerical verification of the high-noise theory. All networks used to create the data in this figure are width-2048, three-layer erf MLPs, trained with MSE loss on noisy MNIST, Fashion-MNIST, and KMNIST with  $N = 4000$ . Main panel: Mean clean test accuracy vs additive-Gaussian-noise strength  $p$  for a nested ensemble of 20 noisy training sets and ten MLPs per noisy set. Error bars show one standard deviation. The blue curve is the mean clean test accuracy of the fitted effective model in Eq. (16). Inset: Output-level comparison at  $p = 0.97$ , using a nested ensemble of 100 noisy training sets and 50 MLPs per noisy set. Each point corresponds to one clean test image and one candidate class, comparing the ensemble-mean network output with the fitted prediction from Eq. (16). The black dashed line in the inset indicates perfect agreement.

#### APPENDIX G: VALIDATION FOR A DEEPER RELU ARCHITECTURE

We repeat the output-level and test-accuracy comparisons we performed in Figs. 4 and 7 for depth-5 ReLU MLPs. All other hyperparameters and evaluation procedures remain unchanged for the respective noise types, and the effective-model coefficients are fitted using the same procedure. As shown in Fig. 10, the centroid-based model continues to describe the ensemble-mean logits and broadly captures the high-noise accuracy curves. These results provide an additional empirical check that the centroid description extends beyond the architecture used in the main text.

#### APPENDIX H: DEPENDENCE OF TEST ACCURACY ON ACTIVATION

In this Appendix, we discuss how the high-noise prediction accuracy depends on the activation function. For analytical

simplicity, we specialize the general result of Appendix C to additive-Gaussian noise. This choice makes the fluctuation covariance particularly transparent, while still capturing the qualitative mechanism expected for general i.i.d. corruption models: The class-dependent signal is controlled by the centroid overlap derived in Eq. (C76), whereas the activation function enters only through architecture-dependent constants.

We use the same data normalization convention as in Appendix C,

$$\frac{1}{d} \sum_{j=1}^d x_{j\alpha} = 0, \quad \frac{1}{d} \sum_{j=1}^d x_{j\alpha}^2 = 1. \quad (\text{H1})$$

Under this normalization, the pure-noise training-training and test-training kernels coincide at the level relevant for this calculation. Thus,

$$S_{\chi^*}^{1 \rightarrow L} = S_{\chi\chi}^{1 \rightarrow L} = S, \quad T_{*\chi}^{1 \rightarrow L} = T_{\chi\chi}^{1 \rightarrow L} = T, \quad \theta_* = \theta_o. \quad (\text{H2})$$

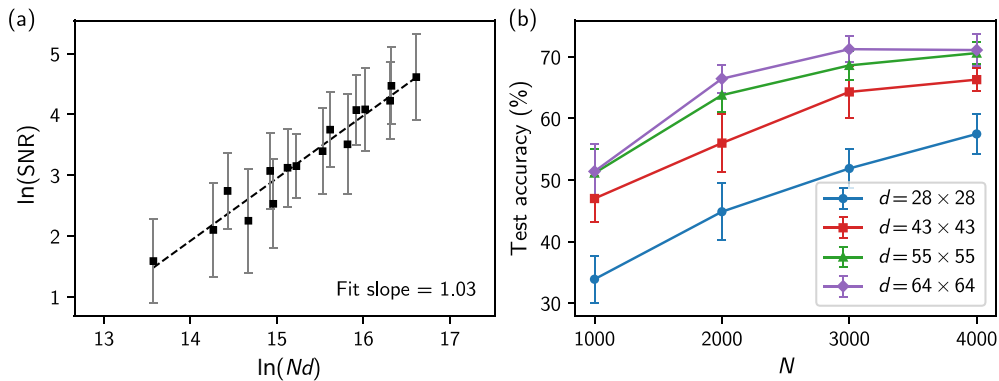


FIG. 8. Empirical  $N, d$  scaling at fixed additive-Gaussian noise probability  $p = 0.97$  and  $C = 10$  classes for width-2048, three-layer erf MLPs trained with MSE loss on MNIST. (a) Mean  $\ln(\text{SNR})$  vs  $\ln(Nd)$  over 500 randomly sampled clean test examples. Per-example SNR is computed from true-class logits averaged over ten initializations per noisy training dataset, using the squared mean divided by the sample variance across 20 noisy training datasets. Squares and error bars show the mean and one standard deviation of  $\ln(\text{SNR})$  across test examples; the dashed line is a linear fit to the means. (b) Mean clean test accuracy vs  $N$  at fixed  $d$ , using predictions from initialization-averaged logits. Means and one-standard-deviation error bars are computed across the 20 noisy training datasets. Larger feature dimensions  $d$  are obtained by resizing MNIST images using bilinear interpolation.

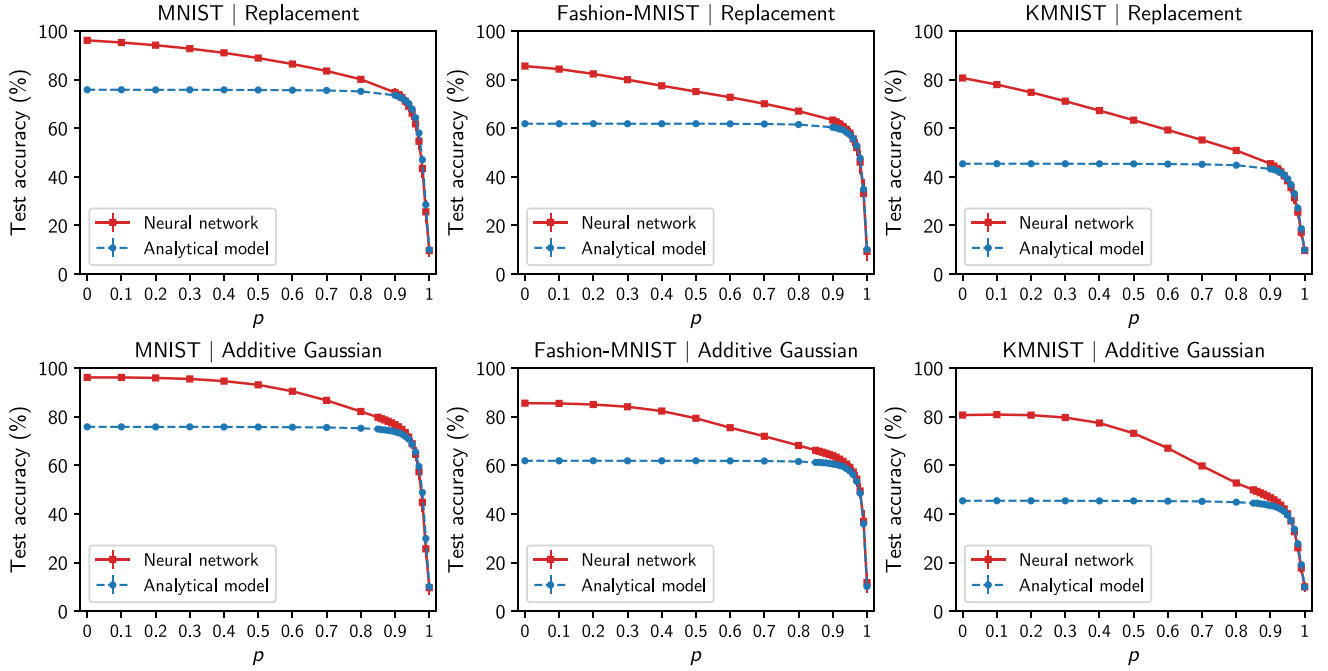


FIG. 9. Clean test accuracy over the full noise range  $p \in [0, 1]$  for replacement noise (top row) and additive-Gaussian noise (bottom row). Columns show MNIST, Fashion-MNIST, and KMNIST, from left to right. Red solid curves show empirical neural-network accuracy; blue dashed curves show the centroid-based effective model of Eq. (16), calibrated at  $p = 0.97$  and evaluated across the full range. Networks are width-2048, three-layer erf MLPs trained with MSE loss,  $N = 4000$ , and  $d = 28 \times 28$ . Ensemble averaging and all other hyperparameters match the accuracy panels of Figs. 4 and 7 for the respective noise types. Error bars show one standard deviation.

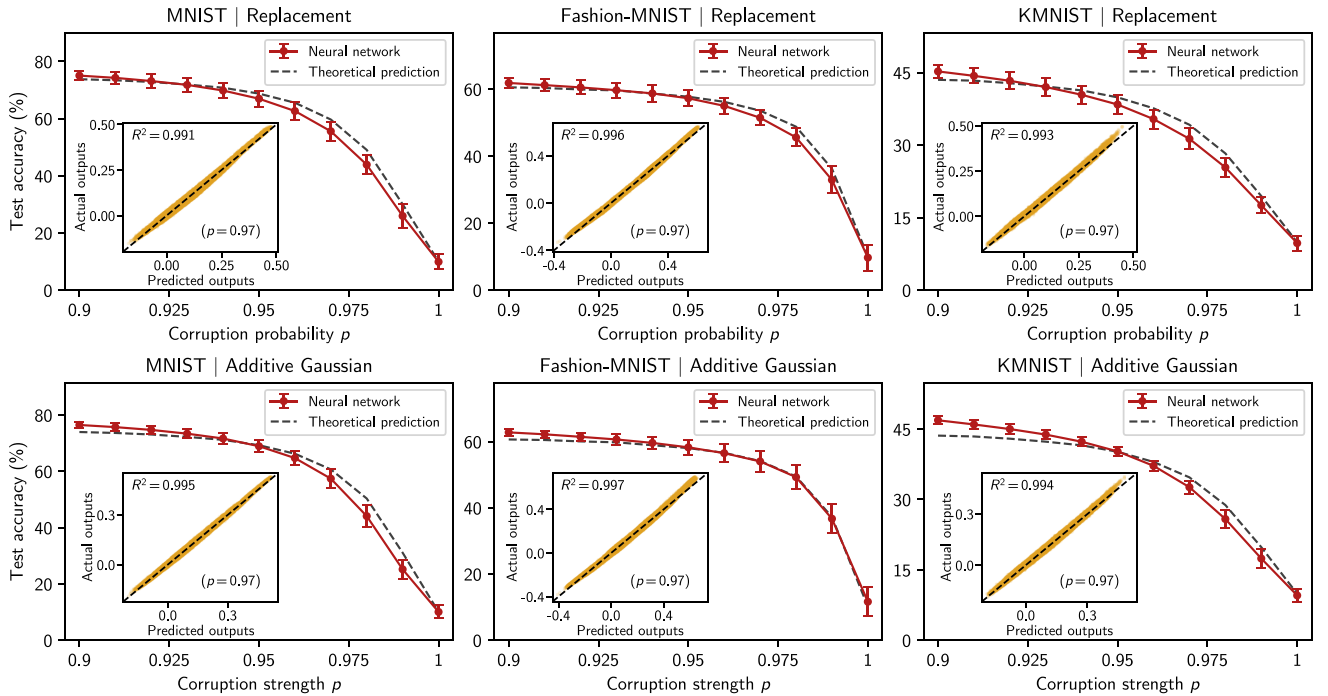


FIG. 10. Comparison with depth-5, width-2048 ReLU MLPs trained with MSE loss under replacement noise (top row) and additive-Gaussian noise (bottom row). Columns show MNIST, Fashion-MNIST, and KMNIST, from left to right. Main panels: Mean clean test accuracy vs  $p$  for the neural networks (red solid curves) and the fitted centroid-based effective model of Eq. (16) (blue dashed curves). Error bars show one standard deviation. Insets: Ensemble-mean output logits vs fitted mean predictions at  $p = 0.97$ , with each point representing a clean test example and candidate class; dashed diagonal lines indicate perfect agreement. Training uses  $N = 4000$  and  $d = 28 \times 28$ . All settings other than depth and activation, including ensemble sizes and fitting procedures, match Figs. 4 and 7 and for the respective noise types.

For additive-Gaussian noise, substituting the corresponding first-layer derivatives into the general high-noise output formula [Eq. (C64)] gives

$$\begin{aligned}
 f_i(x_*; \epsilon) &= \frac{\theta_o \psi}{C} + \sqrt{\frac{N/C}{d}} \frac{\sqrt{V_d}}{\theta_d - \theta_o} \hat{\xi}_i \\
 &+ C_w \frac{\epsilon N/C}{\theta_d - \theta_o} \left( -2S(1 - \theta_o \psi)(1 - 2\theta_o \psi) \right. \\
 &\left. + T \Delta E_\alpha^i \left[ \frac{1}{d} \sum_{j=1}^d x_{j*} x_{j\alpha} \right] \right) \\
 &+ \mathcal{O}(\epsilon N^0) + \mathcal{O}(\epsilon^2) + \mathcal{O}(N^0/\sqrt{d}) \\
 &+ \mathcal{O}(\epsilon N/\sqrt{d}) + \mathcal{O}(N/d), \tag{H3}
 \end{aligned}$$

where

$$\begin{aligned}
 V_d &= C_w^2 \left[ 2S^2(1 - \theta_o \psi)^2 \left( 1 - \frac{4}{C} \theta_o \psi (1 - \theta_o \psi) \right) \right. \\
 &\left. + T^2 \left( 1 - \frac{1}{C} \theta_o \psi (2 - \theta_o \psi) \right) \right]. \tag{H4}
 \end{aligned}$$

Here,  $\hat{\xi}_i$  denotes the Gaussian fluctuation described in Appendix C and  $\Delta E_\alpha^i[\cdot]$  is the class-centered average defined in Eq. (C62). The only term in Eq. (H3) with classifying power is the centroid-overlap term

$$\Delta E_\alpha^i \left[ \frac{1}{d} \sum_{j=1}^d x_{j*} x_{j\alpha} \right]. \tag{H5}$$

All remaining terms are either class-independent shifts or random fluctuations. Under the normalization and unit-variance noise conventions used here, the coefficient multiplying this centroid term has the correct non-negative orientation; equivalently,  $T \geq 0$ , as shown in Appendix D. Thus, different activations change the magnitude of the signal and the fluctuation scale, but they do not reverse the orientation of the leading centroid decision rule in the setting considered here.

For fixed data and noise level, activation-dependent robustness is therefore governed by the noise-to-signal ratio between the fluctuation scale and the coefficient of this centroid term. Equivalently, maximizing prediction accuracy corresponds to minimizing

$$\frac{V_d}{T^2}. \tag{H6}$$

Since the optimal activation should not depend sensitively on the number of classes, it is useful to examine the large- $C$  limit. In this limit,

$$V_d = C_w^2 [2S^2(1 - \theta_o \psi)^2 + T^2]. \tag{H7}$$

Thus, at large  $C$ , the excess fluctuation beyond the unavoidable  $T$  contribution is controlled by

$$S^2(1 - \theta_o \psi)^2. \tag{H8}$$

Consequently, for the generic case in which the pure-noise off-diagonal NTK is nonzero, the large- $N$  relation

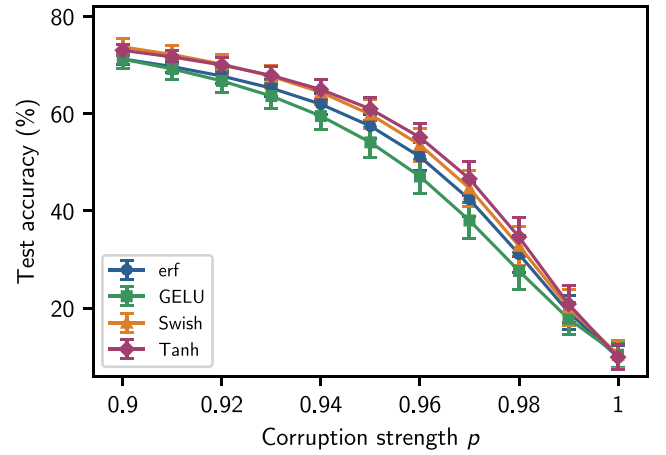


FIG. 11. Mean test accuracy, with standard deviations shown as error bars, for an ensemble of 50 independent three-layer, width-2048 neural networks with four different activations: erf, GELU, Swish, and Tanh. The networks are trained with MSE loss on a noisy subset of 4000 MNIST data points corrupted with additive-Gaussian noise.

$\theta_o \psi = 1 + \mathcal{O}(N^{-1})$  implies that this term is suppressed by  $\mathcal{O}(N^{-2})$ . Thus, a broad class of activations should have nearly identical signal-to-noise ratios, and therefore similar test accuracy. In other words, the leading high-noise classifier is not strongly controlled by the detailed choice of activation; the activation mainly changes subleading architecture-dependent prefactors multiplying the same centroid rule.

Figure 11 tests this prediction numerically. We compare four activations: erf, GELU, Swish, and Tanh. The resulting test-accuracy curves are very similar across the full range of noise probabilities  $p$ . Aside from a modest advantage for Tanh and Swish at the highest noise levels, all four activations attain comparable performance and display nearly the same dependence on  $p$ .

## APPENDIX I: EXPLANATION OF NONMONOTONIC TEST ACCURACY VARIANCE

Here, we show that the nonmonotonic dependence of the test-accuracy variance on the noise probability  $p$  follows directly from the general form of the output logits in Eq. (13). For clarity, we consider the following toy model. The logit of the correct class is

$$l_* = \hat{\xi}_* + a(1 - p), \tag{I1}$$

where  $\hat{\xi}_* \sim \mathcal{N}(0, 1)$  and  $a > 0$  is a constant. The logits of the  $C - 1$  incorrect classes are

$$l_i = \hat{\xi}_i, \quad i = 1, \dots, C - 1, \tag{I2}$$

where each  $\hat{\xi}_i \sim \mathcal{N}(0, 1)$ . We assume all Gaussian variables are independent. This captures the essential structure of Eq. (13): The correct class receives a positive deterministic offset, reflecting its larger centroid overlap, while the incorrect classes do not. For simplicity, we take this offset to be the constant  $a(1 - p)$ , and we normalize the Gaussian fluctuations to zero mean and unit variance.

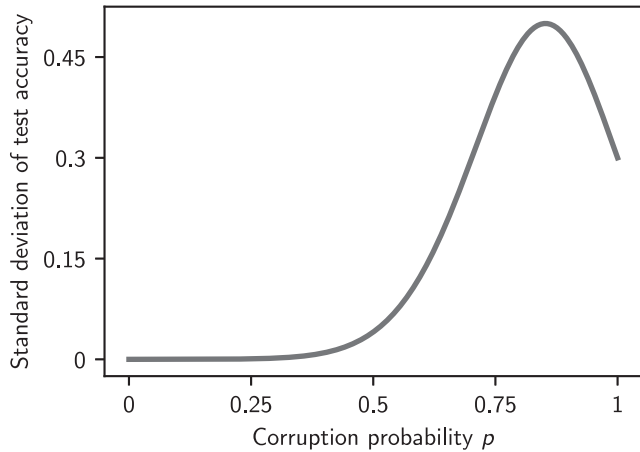


FIG. 12. Standard deviation of the test accuracy of the toy model vs the noise probability  $p$ . We set the constants  $a = 10$  and  $C = 10$ .

For a test input  $x_*$ , define the correctness indicator

$$\text{Acc}(x_*) = 1 \left\{ l_*(x_*) > \max_{1 \leq i \leq C-1} l_i(x_*) \right\}. \quad (13)$$

The population mean test accuracy is therefore

$$\mathbb{E}[\text{Acc}] = \mathbb{E}_{x_*} \left[ 1 \left\{ l_*(x_*) > \max_{1 \leq i \leq C-1} l_i(x_*) \right\} \right]. \quad (14)$$

Since  $\text{Acc}(x_*)$  is an indicator variable,  $\text{Acc}(x_*)^2 = \text{Acc}(x_*)$ , and thus its population variance is

$$\text{Var}[\text{Acc}] = \mathbb{E}[\text{Acc}] - \mathbb{E}[\text{Acc}]^2 = \mathbb{E}[\text{Acc}] (1 - \mathbb{E}[\text{Acc}]). \quad (15)$$

Hence, the variance is maximized when  $\mathbb{E}[\text{Acc}] = 1/2$ , so it is necessarily a nonmonotonic function of the mean accuracy.

It therefore remains only to show that the mean accuracy is monotonic in  $p$ . Conditioning on the correct-class noise  $\hat{\xi}_* = u$ , the probability that the correct logit exceeds all  $C - 1$  incorrect logits is

$$\Phi(u + a(1 - p))^{C-1}. \quad (16)$$

Averaging over  $u \sim \mathcal{N}(0, 1)$  gives

$$\mathbb{E}[\text{Acc}] = \int_{-\infty}^{\infty} du \phi(u) \Phi(u + a(1 - p))^{C-1}, \quad (17)$$

where  $\phi$  and  $\Phi$  are the standard normal probability density function and cumulative distribution function, respectively. Because  $\Phi$  is strictly increasing, this expression is monotonically decreasing in  $p$ . It follows that  $\text{Var}[\text{Acc}]$ , being of the form  $m(1 - m)$  with  $m = \mathbb{E}[\text{Acc}]$ , is generically nonmonotonic in  $p$ . This proves the claim.

Figure 12 shows the predicted standard deviation of the test accuracy in this toy model as a function of  $p$  for  $a = 10$  and  $C = 10$ . The nonmonotonic behavior is clearly reproduced, showing that it arises directly from the competition between a decreasing correct-class offset and Gaussian logit fluctuations.

- 
- [1] D. Angluin and P. Laird, Learning from noisy examples, *Mach. Learn.* **2**, 343 (1988).
  - [2] M. Kearns, Efficient noise-tolerant learning from statistical queries, *J. ACM* **45**, 983 (1998).
  - [3] A. Blum, A. Kalai, and H. Wasserman, Noise-tolerant learning, the parity problem, and the statistical query model, *J. ACM* **50**, 506 (2003).
  - [4] B. Frenay and M. Verleysen, Classification in the presence of label noise: A survey, *IEEE Trans. Neural Networks Learn. Syst.* **25**, 845 (2014).
  - [5] N. Natarajan, I. S. Dhillon, P. Ravikumar, and A. Tewari, Learning with noisy labels, in *Advances in Neural Information Processing Systems 26 (NeurIPS 2013)* (Curran Associates, Inc., Red Hook, NY, USA, 2013), pp. 1196–1204.
  - [6] D. Rolnick, A. Veit, S. Belongie, and N. Shavit, Deep learning is robust to massive label noise, [arXiv:1705.10694](https://arxiv.org/abs/1705.10694).
  - [7] C. Northcutt, L. Jiang, and I. Chuang, Confident learning: Estimating uncertainty in dataset labels, *J. Artif. Int. Res.* **70**, 1373 (2021).
  - [8] T. Emmanuel, T. Maupong, D. Mpoeleng, T. Semong, B. Mphago, and O. Tabona, A survey on missing data in machine learning, *J. Big Data* **8**, 140 (2021).
  - [9] K. Matsuoka, Noise injection into inputs in back-propagation learning, *IEEE Trans. Syst. Man, Cybern.* **22**, 436 (1992).
  - [10] L. Holmstrom and P. Koistinen, Using additive noise in back-propagation training, *IEEE Trans. Neural Networks* **3**, 24 (1992).
  - [11] C. M. Bishop, Training with noise is equivalent to Tikhonov regularization, *Neural Comput.* **7**, 108 (1995).
  - [12] G. An, The effects of adding noise during backpropagation training on a generalization performance, *Neural Comput.* **8**, 643 (1996).
  - [13] J. Sietsma and R. J. Dow, Creating artificial neural networks that generalize, *Neural Networks* **4**, 67 (1991).
  - [14] R. Reed, S. Oh, and R. Marks, Regularization using jittered training data, in *[Proceedings 1992] IJCNN International Joint Conference on Neural Networks* (IEEE, Piscataway, NJ, USA, 1992), Vol. 3, pp. 147–152.
  - [15] R. M. Zur, Y. Jiang, L. L. Pesce, and K. Drukker, Noise injection for training artificial neural networks: A comparison with weight decay and early stopping, *Med. Phys.* **36**, 4810 (2009).
  - [16] O. H. Ramírez-Agudelo Sr, N. Gorea, A. Reif, L. Bonasera, and M. Karl Sr, The role of noisy data in improving CNN robustness for image classification, in *Applications of Machine Learning 2025* (SPIE, Bellingham, WA, USA, 2025), Vol. 13606, pp. 197–212.
  - [17] A. Globerson and S. Roweis, Nightmare at test time: Robust learning by feature deletion, in *Proceedings of the 23rd International Conference on Machine Learning, ICML'06* (Association for Computing Machinery, New York, NY, USA, 2006), pp. 353–360.
  - [18] O. Dekel and O. Shamir, Learning to classify with missing and corrupted features, in *Proceedings of the 25th International Conference on Machine Learning, ICML'08* (Association

- for Computing Machinery, New York, NY, USA, 2008), pp. 216–223.
- [19] L. Van Der Maaten, M. Chen, S. Tyree, and K. Q. Weinberger, Learning with marginalized corrupted features, in *Proceedings of the 30th International Conference on International Conference on Machine Learning*, ICML'13 (JMLR.org, Cambridge, MA, USA, 2013), Vol. 28, pp. I-410–I-418.
- [20] D. Hendrycks, N. Mu, E. D. Cubuk, B. Zoph, J. Gilmer, and B. Lakshminarayanan, AugMix: A simple data processing method to improve robustness and uncertainty, in *Proceedings of the 8th International Conference on Learning Representations (ICLR, 2020)* (OpenReview.net, 2020).
- [21] D. Hendrycks and T. G. Dietterich, Benchmarking neural network robustness to common corruptions and perturbations, *International Conference on Learning Representations (ICLR)* (2019).
- [22] N. Mu and J. Gilmer, MNIST-C: A robustness benchmark for computer vision, [arXiv:1906.02337](https://arxiv.org/abs/1906.02337).
- [23] Y. LeCun, C. Cortes, and C. Burges, MNIST handwritten digit database, ATT Labs [Online], 2010, <http://yann.lecun.com/exdb/mnist>.
- [24] T. Clanuwat, M. Bober-Irizar, A. Kitamoto, A. Lamb, K. Yamamoto, and D. Ha, Deep learning for classical japanese literature, in *NeurIPS 2018 Workshop on Machine Learning for Creativity and Design* (Montréal, Canada, 2018).
- [25] H. Xiao, K. Rasul, and R. Vollgraf, Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms, [arXiv:1708.07747](https://arxiv.org/abs/1708.07747).
- [26] A. Jacot, F. Gabriel, and C. Hongler, Neural tangent kernel: Convergence and generalization in neural networks, in *Advances in Neural Information Processing Systems 31* (NeurIPS 2018) (Curran Associates, Inc., Red Hook, NY, USA, 2018), Vol. 31, pp. 8571–8580.
- [27] R. Tibshirani, T. Hastie, B. Narasimhan, and G. Chu, Diagnosis of multiple cancer types by shrunken centroids of gene expression, *Proc. Natl. Acad. Sci. USA* **99**, 6567 (2002).
- [28] J. Snell, K. Swersky, and R. S. Zemel, Prototypical networks for few-shot learning, in *Advances in Neural Information Processing Systems 30 (NeurIPS 2017)* (2017), pp. 4077–4087.
- [29] S. Guerriero, B. Caputo, and T. Mensink, Deep nearest class mean classifiers, in ICLR Workshop (2018).
- [30] V. Pappayan, X. Y. Han, and D. L. Donoho, Prevalence of neural collapse during the terminal phase of deep learning training, *Proc. Natl. Acad. Sci. USA* **117**, 24652 (2020).
- [31] M. E. A. Seddik and M. Tamaazousti, Neural networks classify through the class-wise means of their representations, *Proc. AAAI Conf. Artif. Intell.* **36**, 8204 (2022).
- [32] V. Kothapalli, Neural collapse: A review on modelling principles and generalization, *Transactions on Machine Learning Research* (2023).
- [33] S. Mei and A. Montanari, The generalization error of random features regression: Precise asymptotics and the double descent curve, *Commun. Pure Appl. Math.* **75**, 667 (2022).
- [34] S. Goldt, B. Loureiro, G. Reeves, F. Krzakala, M. Mézard, and L. Zdeborová, The Gaussian equivalence of generative models for learning with shallow neural networks, in *Proceedings of the 2nd Mathematical and Scientific Machine Learning Conference* (PMLR, Cambridge, MA, USA, 2022), Vol. 145, pp. 426–471.
- [35] H. Hu and Y. M. Lu, Universality laws for high-dimensional learning with random features, *IEEE Trans. Inf. Theory* **69**, 1932 (2023).
- [36] A. Montanari and B. N. Saeed, Universality of empirical risk minimization, in *Proceedings of the 35th Conference on Learning Theory (COLT)* (PMLR, Cambridge, MA, USA, 2022), Vol. 178, pp. 4310–4312.
- [37] Y. Bahri, B. Hanin, A. Brossollet, V. Erba, C. Keup, R. Pacelli, and J. B. Simon, Les Houches lectures on deep learning at large and infinite width, *J. Stat. Mech.* (2024) 104012.
- [38] D. A. Roberts, S. Yaida, and B. Hanin, *The Principles of Deep Learning Theory* (Cambridge University Press, Cambridge, MA, USA, 2022), Vol. 46.
- [39] R. Price, A useful theorem for nonlinear devices having Gaussian inputs, *IRE Trans. Inf. Theory* **4**, 69 (1958).
- [40] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions* (Dover, New York, 1965).
- [41] F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller, B. V. Saunders, H. S. Cohl, and M. A. McClain (eds.), NIST Digital Library of Mathematical Functions, Chap. 18, Release 1.2.8 (September 15, 2026), <https://dlmf.nist.gov/>.