Asymmetric Scaling Laws from Sparse Features

John Sous
Yale University
john.sous@yale.edu

Michael Winer*
Institute for Advanced Study
mikewins@ias.edu

Abstract

We introduce a model for neural scaling laws under sparse activations. In the model, test loss is often dominated by rare coordinates that are never observed in the training input. This mechanism induces a novel bottleneck absent from dense models. We derive the asymptotic population loss in both the underparameterized and overparameterized regimes, and show that the loss exhibits a double-descent peak near the interpolation threshold—where the number of parameters is just sufficient to fit the training data—resulting in a loss curve governed by two distinct scaling exponents—one for each regime—with a gap determined by the degree of sparsity. Additionally, we derive a compute-optimal frontier that favors increasing dataset size over model capacity under fixed compute budgets. We also analyze gradient-descent dynamics and identify a scaling law for the probability that fixed-step gradient descent becomes unstable. We further show that the sparsity-induced effect persists under nonlinear activations. Finally, on the Pile dataset, we show that sparsity in language data turns data coverage into a limiting resource, as our theory predicts. Code is available at asym-scaling.

1 Introduction

Scaling laws describe predictable relationships between model performance and key resource variables: model size (number of parameters) N, dataset size (number of training examples or tokens) D, and compute budget C. These empirical laws provide crucial guidance for efficiently allocating resources when training large machine learning models, particularly large language models (LLMs). Foundational work by Kaplan et al. showed that test loss ℓ decreases as a power law with respect to N and D when scaling one while holding the other fixed [1]. Building on this, Hoffmann et al. showed that under the constraint of a fixed total compute budget $C \propto ND$, both N and D should be scaled jointly in proportion to minimize ℓ [2]. This leads to a compute-optimal scaling law that offers a principled prescription for balancing model and data scale in large-scale training.

Despite intense research, a comprehensive theoretical understanding of scaling laws remains elusive. One approach to this challenge models the data as Gaussian with power-law covariance, applies a random embedding into a lower-dimensional representation, and uses linear regression to characterize the resulting scaling laws [3–6]. This framework [6] has been shown, both theoretically and empirically, to produce the scaling law $\ell \propto \left(\frac{1}{N} + \frac{1}{D}\right)^{\alpha}$ and the "Chinchilla" compute-optimal allocation in which, under a compute budget $C \propto ND$, the optimal choice scales as $N^*(C) \asymp D^*(C) \asymp C^{1/2}$.[2]

An important aspect not determined by a framework that yields a compute-optimal exponent close to $\frac{1}{2}$ is whether the power-law exponent governing loss decay with model size, α_N, matches that governing decay with data size, α_D, as empirical studies suggest need not be the case [2].[3] In this

*Present affiliation: Alignment Research Center; mike@alignment.org

[2]This compute-allocation law arises in this framework both in the Bayes-optimal [5, 6] and one-pass SGD [7, 8] settings. In the Bayes-optimal case, this scaling can be found by analytically extremizing the loss under the compute constraint. Results based on random matrix theory support the same scaling in one-pass stochastic gradient descent (SGD) dynamics.

[3]Frameworks that predict $\alpha_N = \alpha_D = \alpha$ necessarily imply $N^*(C) \asymp D^*(C) \asymp C^{1/2}$ under the fixed-compute constraint $C \propto ND$. However, a small asymmetry between α_N and α_D can still yield compute-optimal exponents close to $\frac{1}{2}$. Thus, near-square-root compute-optimal scaling does not by itself imply symmetric scaling.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

work, we propose a simple theoretical model that reproduces the observed scaling asymmetry and naturally explains the emergence of two distinct exponents in the population loss. Our model is a random embedding followed by a linear readout, with a sparse input activation structure such that only a subset of coordinates of $\mathbf{x}$ is active. This setup is motivated by the idea that the data-generating process excites only a sparse subset of high-dimensional feature directions. The resulting loss displays distinct scaling with N and D. We derive an asymptotic scaling law in which the difference between α_N and α_D arises continuously as the sparsity level is varied.[4]

Our main contributions are as follows.

- We introduce the notion of scaling laws for sparse activations by introducing a model that captures the impact of sparsity on optimization and resulting scaling behavior.
- We derive the population loss for the sparse random feature model, yielding a two-exponent scaling law with an intrinsic asymmetry between the underparameterized and overparameterized regimes and a double-descent peak near the interpolation threshold [9–15].
- We derive the compute-optimal frontier under a fixed compute budget, and show that increasing sparsity improves compute-efficiency while shifting the optimal allocation toward larger datasets.
- We analyze the training dynamics and convergence properties of the loss during optimization, deriving a scaling law for the failure of fixed-step gradient descent (GD).
- We experimentally verify in a nonlinear two-layer network that scaling asymmetry continues to arise from sparsity rather than from nonlinearity.
- On the Pile dataset [16], we show that the double-descent peak shifts from $D = N$ to larger D, as our theory predicts, and that the asymmetry $\alpha_N - \alpha_D$ is larger than in a dense baseline.

Related Work. Empirical scaling laws show power-law loss improvement with model size and data, and motivate compute-optimal training rules [1, 2]. On the theory side, *solvable* models based on random features, kernels, and high-dimensional regression derive closed-form or deterministic-equivalent scaling predictions [3–6, 17, 18]. Complementary *dynamical* models study protocol- and time-dependent scaling (*e.g.*, one-pass vs. multi-pass training) and can exhibit multiple scaling phases [7, 8, 19]. In contrast, we focus on sparse feature activation, and show it can *intrinsically* yield different exponents in the model-limited versus data-limited regimes even in Bayes-optimal learning. We thus isolate rare feature coverage, rather than optimizer dynamics or feature learning, as a distinct source of scaling asymmetry (Section 3.4). See Appendix A for more discussion of related works.

2 Statement of Problem

In this section, we motivate the model and outline the main goals of our analysis.

2.1 Learning under Sparse High-Dimensional Data

We first specify the data-generation process that yields a sparse power-law structure in the inputs, then describe the random feature model used to learn from these high-dimensional inputs.

2.1.1 Data Generation Process: Bernoulli-Random Activations with Power-Law Covariance

We consider input data $\mathbf{X} = [\,\mathbf{x}_1, \ldots, \mathbf{x}_D\,] \in \mathbb{R}^{M \times D}$, with each $\mathbf{x}_d \in \mathbb{R}^M$, where D is the number of training examples (data size) and M is the data dimension. We henceforth drop the d subscript and simply refer to a single training example as $\mathbf{x}$. We consider a structured data-generation process in which each input coordinate of $\mathbf{x} \in \mathbb{R}^M$ is randomly activated according to a heavy-tailed sparsity pattern. Specifically, for $j = 1, 2, \ldots, M$, the jth coordinate of $\mathbf{x}$ is drawn as

$$\mathbb{P}(x_j = 0) = 1 - j^{-\alpha_1 - 1}, \qquad \mathbb{P}\Big(x_j = \pm j^{-(\alpha_2+1)/2}\Big) = \tfrac{1}{2}\, j^{-\alpha_1 - 1}. \tag{1}$$

Hence $\mathbb{E}[x_j] = 0$ and $\mathrm{Var}(x_j) = j^{-\alpha_1 - \alpha_2 - 2}$. Let $p_j := \mathbb{P}(x_j \neq 0) = j^{-\alpha_1 - 1}$. We require $\alpha_1 \geq -1$ to ensure that $p_j \leq 1$. Additionally, to ensure that the target variance is finite, the sum over the coordinate-wise variances $\sum_{j=1}^{\infty} j^{-\alpha_1 - \alpha_2 - 2}$ must converge, which holds iff $\alpha_1 + \alpha_2 + 1 > 0$.

This formulation introduces sparsity through Bernoulli input activations: most coordinates of $\mathbf{x}$ are zero, while the few active ones follow a heavy-tailed scaling controlled by α_1 and α_2.

[4] We note that in training scenarios such as few-pass SGD, the underlying sparsity may be partially or entirely obscured. This disparity suggests a novel phase transition as the number of epochs increases.

2.1.2 Random Feature Model

We study a linear model built from $N \ll M$ features randomly embedded from these inputs, with $M \gg \max\{N, D\}$ so that truncation does not affect the power-law tail asymptotics. Each input $\mathbf{x} \in \mathbb{R}^M$ is mapped to features $\boldsymbol{\phi}$ via a fixed random embedding $\mathbf{u} \in \mathbb{R}^{N \times M}$, with $u_{ij} \sim \mathcal{N}(0, 1/N)$ and its $N \times M$ elements picked i.i.d., $\boldsymbol{\phi} = \mathbf{ux}$, and the (per-example) prediction is $\hat{y}(\mathbf{x}) = \boldsymbol{\theta}^\top \boldsymbol{\phi} = \boldsymbol{\theta}^\top \mathbf{ux}$ (and the batched prediction is $\hat{\mathbf{y}} = \boldsymbol{\theta}^\top \boldsymbol{\Phi} = \boldsymbol{\theta}^\top \mathbf{uX}$), with trainable $\boldsymbol{\theta} \in \mathbb{R}^N$.[5] The target function is $y(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} = \sum_{j=1}^M w_j x_j$, for random $\mathbf{w} \in \mathbb{R}^M$ with w_j i.i.d. and $\mathrm{Var}(w_j) = 1$.

This generative model allows us to examine how sparsity and heavy-tailed structure in the data influence performance. As we show below, these structural properties induce a loss scaling law with two distinct exponents, $\alpha_N \neq \alpha_D$, a sparsity-dependent compute-optimal frontier with an exponent α_C that shifts the optimal allocation toward data, and a regime in which GD fails to converge. The resulting regimes are summarized in Figure 1. This scaling asymmetry also persists in experiments with nonlinear MLPs.

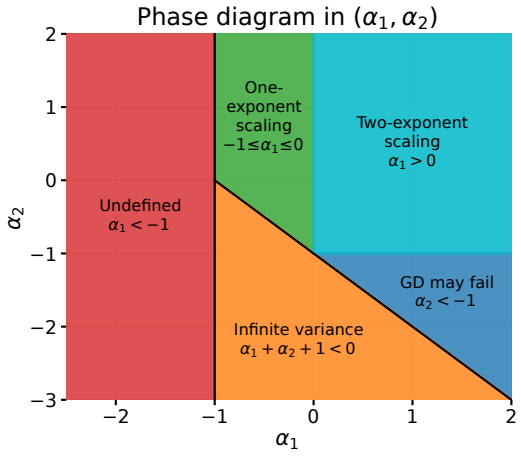

Figure 1: **Phase diagram in** (α_1, α_2)**.** Solid black lines mark the boundary beyond which the model is well defined. Within it we identify three regimes: symmetric one-exponent scaling, asymmetric two-exponent scaling, and a GD-failure regime with its own scaling law.

Interpretation: High-Dimensional Representations with Rare but Informative Features. Sparse activations of $\mathbf{x}$ can be understood as rare but highly informative features that appear only in a small fraction of examples yet carry disproportionate predictive value, *e.g.*, an extremely infrequent keyword such as "pheochromocytoma," rare financial shocks such as the 2008 crisis, or an uncommon diagnostic biomarker.

The random feature model can be motivated by interpreting the high-dimensional vector $\mathbf{x}$ as a nonlinear expansion of a much lower-dimensional latent variable $\tilde{\mathbf{x}}$ [24]. The components of $\mathbf{x}$ could correspond to all polynomial or Fourier basis functions of $\tilde{\mathbf{x}}$, while the features $\boldsymbol{\phi} = \mathbf{ux}$ represent the smaller subset of directions actually captured by our model. Although the mapping through $\mathbf{u}$ is entirely linear, introducing randomness, it can be interpreted as the first layer of a two-layer network (with no nonlinearity) whose weights are frozen at random initialization [25]. In this view, the model explores how much predictive structure can be recovered when learning is restricted to a fixed, randomly oriented subspace of the high-dimensional representation.

Thus, this random feature model with sparse activations reveals how rare, high-impact signals shape optimization and give rise to distinct scaling laws, highlighting how sparsity fundamentally alters achievable performance.

2.2 Main Goals

Our goal is to characterize the trained-to-completion loss for the random sparse feature model described above. The population loss of the trained estimator is (Appendix B)

$$\ell_{\text{Bayes}} = \mathbb{E}_{\mathbf{x}}\left[(y - \hat{y}(\mathbf{x}))^2\right] = \mathbb{E}_{\mathbf{x}}\left[\left(\mathbf{w}^\top \mathbf{x} - \boldsymbol{\theta}^{\star\top} \mathbf{ux}\right)^2\right], \tag{2}$$

where $\boldsymbol{\theta}^\star$ denotes the minimum-ℓ_2-norm empirical-risk minimizer, equivalently the solution reached by GD from zero initialization when the empirical loss is trained to completion. Note that $\hat{y}(\mathbf{x}) = \boldsymbol{\theta}^{\star\top} \mathbf{ux}$ corresponds to a linear function of $\mathbf{x}$ whose weight vector lies in the row space of $\mathbf{u}$.

[5] Our model is, in effect, a sketched linear regression. It is random-feature-like in the sense of a frozen random map followed by a trained linear readout, but it differs from canonical random-features models, *e.g.* [12, 20, 21], which study nonlinear feature maps in high-dimensional regimes. Extending those frameworks to the sparse, strongly non-Gaussian/heavy-tailed designs studied here would require going beyond the Gaussian-equivalence tools used in the nonlinear random-features literature [14, 22, 23].

Consequently, when $\mathbf{w} \notin \mathrm{rowspan}(\mathbf{u})$, this feature map cannot recover $y = \mathbf{w}^\top \mathbf{x}$ exactly, and a nonzero approximation error is unavoidable.

We aim to derive the scaling laws and compute-optimal frontier for this Bayes-optimal loss, emphasizing the dependence on the data distribution and its effect on optimization. In particular, we show that a two-exponent loss scaling law and a sparsity-dependent compute-optimal exponent can arise purely from the structure of the input distribution, in which $\mathbf{x}$ contains sparse but informative coordinates. Our objective is to identify exponents α_N, α_D, and α_C such that $\ell_{\text{Bayes}} \asymp N^{-\alpha_N} + D^{-\alpha_D}$, and the compute-optimal frontier satisfies $\ell^\star_{\text{Bayes}} \asymp C^{-\alpha_C}$, where $\ell^\star_{\text{Bayes}}$ is the loss achieved by compute-optimal allocation. We also study GD in this setting, examining how optimization dynamics interact with the random feature map and the sparse data distribution, and whether it attains the Bayes rates or instead exhibits additional computational constraints, which we dub a scaling law of failure of GD. All proofs are deferred to the appendices.

3 Scaling Asymptotics of Sparse Random Features

Our strategy is to analyze the scaling of the Bayes-optimal loss, Eq. (2), for our model of sparse random features (specified in Section 2.1) in the *underparameterized* regime, where N is much smaller than D and in the *overparameterized* regime, where N vastly exceeds D. We will show that these regimes are governed by two distinct power laws, with different exponents, demonstrating that the loss is asymmetric and thus follows a two-exponent scaling law. This symmetry is broken precisely due to the sparsity of the input activation pattern. See Appendix C for details.

3.1 Loss Scaling from Unmodeled Features

The random feature model reduces to linear regression in the effective weight vector $\hat{\mathbf{w}} = \mathbf{u}^\top \boldsymbol{\theta}$. The population mean-squared error can be expressed as $\ell(\hat{\mathbf{w}}) = \mathbb{E}_{\mathbf{x}}\big[(\mathbf{w}^\top \mathbf{x} - \hat{\mathbf{w}}^\top \mathbf{x})^2\big] = (\mathbf{w} - \hat{\mathbf{w}})^\top \boldsymbol{\Sigma}_x (\mathbf{w} - \hat{\mathbf{w}})$, where $\boldsymbol{\Sigma}_x = \mathbb{E}[\mathbf{x}\mathbf{x}^\top]$ is diagonal since the input coordinates are independent. To expose the scaling mechanism, consider a predictor that matches the first k coordinates of $\mathbf{w}$ and sets all remaining coordinates to zero: $\hat{w}_j = w_j \mathbf{1}_{\{j \le k\}}$. Then $\mathbf{w} - \hat{\mathbf{w}}$ has entries 0 for $j \le k$ and w_j for $j > k$, and substituting into the expression for $\ell(\hat{\mathbf{w}})$ yields $\ell(\hat{\mathbf{w}}) = \sum_{j>k} w_j^2 \ \mathrm{Var}(x_j)$. Under our assumptions $w_j^2 = O(1)$, the population loss is therefore governed, up to constants, by the *unmodeled input variance* $\sum_{j>k} \mathrm{Var}(x_j)$. This calculation suggests that the asymptotic behavior of the loss is governed by the tail of the variance spectrum of the input distribution:

$$\ell_{\text{Bayes}} \asymp \sum_{j>k} \mathrm{Var}(x_j). \tag{3}$$

The argument above is only an intuition-building asymptotic device to make the mechanism transparent. Appendix C, particularly Section C.2.2 (together with Proposition 3.1 and its proof in Section C.2.1), provides a rigorous proof.

3.2 Prior Results: Scaling Law for Uniformly Activated Data with Power-Law Covariance

Before analyzing our model with sparse activations, we first review the scaling laws established for random feature models with fully active (non-sparse) input coordinates [6]. These results form the baseline against which the effects of sparsity will be contrasted.

Uniformly Activated Data with Power-Law Covariance. In the original formulation of Maloney et al. [6], every input coordinate is active and is modeled as an independent Gaussian variable with variance decaying as a power law: $x_j \sim \mathcal{N}(0,\, j^{-\alpha-1}), \qquad j = 1, \ldots, M, \quad \alpha > 0$. This matches our model's fully activated covariance structure under the choice $\alpha_1 = -1$ and $\alpha_2 = \alpha$.

Symmetry between Underparameterized and Overparameterized Regimes. For this non-sparse data structure, the optimal-loss scaling law was analyzed in detail in [6]. We produce a simple intuitive argument showing why, in this setting, a *single* exponent governs the asymptotic behavior of the loss in both the under- and overparameterized regimes, yielding a symmetric one-exponent law $\ell_{\text{Bayes}} \propto \left(\frac{1}{N} + \frac{1}{D}\right)^\alpha$ with $\alpha_N = \alpha_D = \alpha$. As explained above, the random feature model reduces to linear regression in the effective weight vector. Thus, if a learned predictor resolves the leading $k \ll M$ coefficients of $\mathbf{w}$, the residual (unmodeled) variance is $\sum_{j>k} j^{-\alpha-1} \asymp \int_k^\infty j^{-\alpha-1}\, dj =$

$\frac{1}{\alpha}\, k^{-\alpha}$. With finitely many parameters N and finitely many samples D, the model can identify at most $O(\min\{N, D\})$ coefficients, since this minimum represents the true bottleneck. This yields the scaling $\ell_{\text{Bayes}} \sim \min\{N, D\}^{-\alpha}$, so that the exponents satisfy $\alpha_N = \alpha_D = \alpha$. In the sparse model of Section 2.1, one still gets a symmetric one-exponent scaling law when $-1 \leq \alpha_1 \leq 0$. In this regime, with high probability the low-index coordinates are activated sufficiently often across D samples that sparsity is effectively subdominant. The scaling is therefore still controlled by $\min\{N, D\}$, as in the dense case, yielding $\alpha_N = \alpha_D = \alpha_1 + \alpha_2 + 1$ (Appendix C.1).

3.3 Two-Exponent Scaling Law for Sparse Activations ($\alpha_1 > 0$)

We now analyze our model of sparse activations for $\alpha_1 > 0$ (Appendix C.2). We begin by analyzing the asymptotic behavior of the loss in the underparameterized regime, where we will show that sparsity is effectively hidden. We then turn to the overparameterized regime, where sparsity becomes consequential and leads to a distinct asymptotic loss behavior. We work in the joint limit $N, D \to \infty$, and we consider asymptotic regimes corresponding to different relative scaling of N and D.[6]

3.3.1 Underparameterized Regime ($N \ll D$).

In this regime, the model fits only the leading high-variance coordinates of $\mathbf{w}$, and the resulting scaling law coincides with that of the non-sparse case.

Proposition 3.1 (Underparameterized Scaling)**.** *Under the sparse activation model described in Section 2.1, the Bayes-optimal loss in the underparameterized regime ($N \ll D$) satisfies*

$$\ell_{\text{Bayes},N} \sim N^{-(\alpha_1+\alpha_2+1)}, \quad \textit{so that } \alpha_N = \alpha_1 + \alpha_2 + 1. \tag{4}$$

Interpretation. The model captures only the leading high-variance coordinates of $\mathbf{w}$. Since sparse coordinates are too rarely observed to influence estimation at this scale, they do not affect the loss. Consequently, the exponent $\alpha_N = \alpha_1 + \alpha_2 + 1$ coincides with the non-sparse case, so sparsity is effectively masked in the underparameterized regime.

3.3.2 Overparameterized Regime ($D \ll N$).

A key quantity in this regime is the number of input coordinates that are ever observed (*i.e.*, activated at least once). We therefore begin by quantifying the effective number of learnable coordinates as a function of D, which will set the correct scale separating the two asymptotic regimes.

Lemma 3.2 (Learnable Coordinates in the Sparse Model when $\alpha_1 > 0$)**.** *Under the sparse activation distribution $p_j = \mathbb{P}(x_j \neq 0) = j^{-\alpha_1-1}$, the expected number of coordinates that are active in at least one of the D training samples is asymptotic to the scale*

$$K(D) = \Gamma\left(1 - \frac{1}{\alpha_1 + 1}\right) D^{\frac{1}{\alpha_1+1}},^{7} \tag{5}$$

where $\Gamma(\cdot)$ is the Gamma function. Moreover, $K(D)$ sets the scale of the number of coefficients of $\mathbf{w}$ that can be estimated from data: a coordinate that is never activated carries no information about its associated weight, and hence cannot contribute to the learned predictor.

Theorem 3.3 (Overparameterized Scaling)**.** *Under the sparse activation model with $\alpha_1 > 0$ and coordinate variances $\mathrm{Var}(x_j) = j^{-\alpha_1-\alpha_2-2}$, the Bayes-optimal loss in the overparameterized regime satisfies*

$$\ell_{\text{Bayes},D} \asymp D^{-\frac{\alpha_1+\alpha_2+1}{\alpha_1+1}}, \quad \textit{so that } \alpha_D = \frac{\alpha_1 + \alpha_2 + 1}{\alpha_1 + 1}. \tag{6}$$

A rigorous proof of Theorem 3.3 is given in Appendix C.2.2 via a continuous argument.

Remark 3.4 (Intuition). A simple heuristic recovers the same sample-limited exponent as in Theorem 3.3. Coordinate j is active in a single sample with probability $p_j = j^{-\alpha_1-1}$, so across D samples the expected number of activations is Dp_j. A coordinate is observed often enough to be

[6] In the sparse activation model, the relevant comparison is ultimately between N and the number of input coordinates ever observed in the dataset, $K(D)$ (quantified below), which grows sublinearly in D when $\alpha_1 > 0$.

[7] When $\alpha_1 > 0$, $K(D) \asymp D^{1/(\alpha_1+1)} = o(D)$, *i.e.* only a sublinear number of coordinates are ever observed. Interpolation is still possible since the data lie in a $K(D)$-dimensional subspace, which a linear predictor in $\mathbb{R}^N$ can fit. Unobserved coordinates receive no signal and are typically suppressed by mild regularization.

estimable once $Dp_j \gtrsim 1$, which yields the cutoff $j \lesssim j_D \equiv D^{1/(\alpha_1+1)}$. Thus the number of learnable coordinates scales as $K(D) \asymp j_D$, in agreement with Lemma 3.2. Coordinates with $j > j_D$ are effectively unobserved. Their contribution to the test error is the unmodeled variance $\sum_{j>j_D} j^{-\alpha_1-\alpha_2-2} \asymp j_D^{-(\alpha_1+\alpha_2+1)} = D^{-\frac{\alpha_1+\alpha_2+1}{\alpha_1+1}}$. Consequently, $\ell_{\mathrm{Bayes},D} \asymp D^{-\frac{\alpha_1+\alpha_2+1}{\alpha_1+1}}$, with $\alpha_D = \frac{\alpha_1+\alpha_2+1}{\alpha_1+1}$, matching the rigorous sample-limited exponent in Theorem 3.3.

Corollary 3.5 (Asymmetry of Exponents)**.** *For all $\alpha_1 > 0$ and $\alpha_1 + \alpha_2 + 1 > 0$, the exponents in the underparameterized and overparameterized regimes satisfy*

$$\alpha_D = \frac{\alpha_1 + \alpha_2 + 1}{\alpha_1 + 1} < \alpha_1 + \alpha_2 + 1 = \alpha_N. \tag{7}$$

Thus, for $\alpha_1 > 0$, the sparse model exhibits an intrinsic asymmetry between the sample-limited and parameter-limited regimes, giving rise to the two-exponent scaling law.

Together, Proposition 3.1 and Theorem 3.3 yield:

$$\boxed{\ell_{\mathrm{Bayes}}(N, D) \asymp N^{-(\alpha_1+\alpha_2+1)} + D^{-(\alpha_1+\alpha_2+1)/(\alpha_1+1)}.} \tag{8}$$

This expression makes the asymmetry explicit: model-limited error decays sharply, whereas data-limited error decays slowly. Since the mechanism depends on sparsity and variance profiles, we expect it to extend beyond Bernoulli masks to broader sparse distributions. Conversely, no asymmetry arises from the covariance alone: a design with independent Gaussian coordinates of the same variances $j^{-(\alpha_1+\alpha_2+2)}$ has every coordinate nonzero in every sample, so no coverage bottleneck forms and, for our flat target without label noise, the symmetric law of Section 3.2 applies with $\alpha = \alpha_1 + \alpha_2 + 1$. Sparsity enters through the activation probabilities p_j, which the covariance fixes only through the product $p_j\, j^{-(\alpha_2+1)}$.

3.4 Relation to Other Sources of Asymmetry

Asymmetric exponents also arise in other solvable models, through different mechanisms. For random-feature regression trained to completion (empirical risk minimization), Defilippis et al. [17] considered a covariance with power-law eigenvalues k^{-a} and a target with energy $k^{-(1+2ar)}$ along eigendirection k, with r the source exponent, and found that, without label noise and at vanishing ridge, the loss decays as $N^{-2a\min(r,1/2)}$ and $D^{-2a\min(r,1)}$; the two exponents therefore differ only for $r > 1/2$, and then the data exponent is the larger. Without a source condition, that is, for w_j i.i.d. as in our model, $2ar = a - 1$ and both exponents are $a - 1$, which is the symmetric law of Section 3.2 [6] with $a = \alpha + 1$, and the dense baseline against which we compare in our experiments. One-pass training is asymmetric as well [7, 8, 19]. Both bottlenecks are then the same spectral tail, cut off at different modes. Width resolves $k \asymp N$ modes, while gradient flow resolves the modes with $k^{-a}t \gtrsim 1$, i.e. $k \asymp t^{1/a}$, and with each sample seen once $t \asymp D$; for a target with energy k^{-s} beyond mode k, the loss scales as N^{-s} in width and as $D^{-s/a}$ in data, so $\alpha_N/\alpha_D = a$. When training is run to completion on reused data, the second cutoff becomes $k \asymp D$ rather than $D^{1/a}$, and the scaling law reverts to the symmetric empirical-risk law above. The asymmetry of Corollary 3.5 instead arises at convergence, for a flat target, because only $K(D) \asymp D^{1/(\alpha_1+1)}$ coordinates are ever observed, which is why the model exponent is the larger, $\alpha_N > \alpha_D$, unlike in the source-condition case.

4 Empirical Study of the Scaling Laws

We now empirically validate the theoretical predictions of the previous section by evaluating the population loss over a wide range of D and N in the sparse random-feature model of Section 2.1. Computational details are provided in Appendices B and F.

4.1 Scaling Collapse and Universality

In many models, including ours and others such as the Chinchilla scaling law [2], the test loss exhibits distinct scaling in two regimes: $\ell \sim D^{-\alpha_D}$ in the data-limited regime and $\ell \sim N^{-\alpha_N}$ in the parameter-limited regime. A natural question is how to interpolate between these two limits. A common ansatz is that the loss takes the form $\ell = \bar{L}(D^{-\alpha_D}, N^{-\alpha_N})$ for some *scale-invariant* function $\bar{L}$ satisfying $\bar{L}(cL_1, cL_2) = c\bar{L}(L_1, L_2)$. Simple choices include $\bar{L}(L_1, L_2) = \max(L_1, L_2)$

or $L_1 + L_2$, but more refined analyses [6, 26] show that $\bar{L}$ often exhibits a *peak* near $L_1 \approx L_2$, a phenomenon known as *double descent* [10, 12].[8] Using the scale invariance of $\bar{L}$, we may write $\ell \cdot N^{\alpha_N} = \bar{L}\left(\frac{D^{-\alpha_D}}{N^{-\alpha_N}}, 1\right)$, so the rescaled loss $\ell \cdot N^{\alpha_N}$ becomes a universal function of the compute ratio $\Xi := \frac{D^{\alpha_D}}{N^{\alpha_N}}$. We therefore expect the rescaled loss curves from different (D, N) to collapse onto a single curve $\mathcal{S}_{\alpha_1,\alpha_2}(\Xi)$: a dimensionless scaling function of the rescaled compute ratio.

Empirical Scaling Collapse. In Figure 2, we find that the full loss curve across a wide range of (D, N) pairs collapses onto a single universal function after appropriate rescaling: $\ell_{\text{Bayes}} \cdot N^{\alpha_1+\alpha_2+1} = \mathcal{S}_{\alpha_1,\alpha_2}\left(\frac{1}{N} D^{\frac{1}{\alpha_1+1}}\right)$, where $\mathcal{S}(u) = \bar{L}(u^{-\alpha_N}, 1)$, so that $\bar{L}(\frac{D^{-\alpha_D}}{N^{-\alpha_N}}, 1) = \mathcal{S}_{\alpha_1,\alpha_2}(\xi)$, with $\xi := \left(\frac{D^{\alpha_D}}{N^{\alpha_N}}\right)^{1/\alpha_N} = \frac{D^{1/(\alpha_1+1)}}{N}$. This implies that knowing the loss curve for one setting of (D, N) suffices to predict its shape at other scales via a simple rescaling transformation. This collapse supports the two-regime scaling theory and recovers the predicted asymptotic behaviors: in the underparameterized regime, $\ell \sim N^{-\alpha_N}$ with $\alpha_N = \alpha_1 + \alpha_2 + 1$, while in the overparameterized regime, $\ell \sim D^{-\alpha_D}$ with $\alpha_D = (\alpha_1 + \alpha_2 + 1)/(\alpha_1 + 1)$.

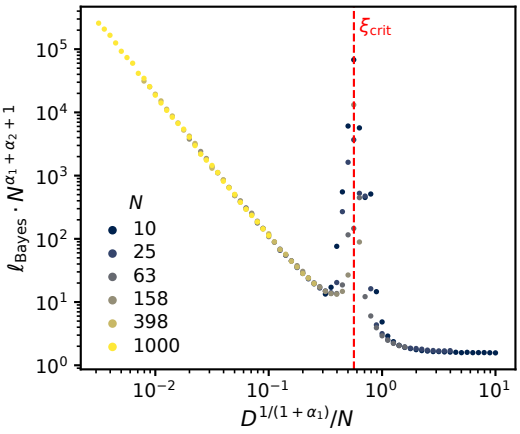

Figure 2: **Scaling collapse and double descent.** We plot the rescaled loss $\ell_{\text{Bayes}} \cdot N^{\alpha_N}$ as a function of $(D^{\alpha_D}/N^{\alpha_N})^{1/\alpha_N}$, across a wide range of values for D, N, for fixed $(\alpha_1 = 1.0, \alpha_2 = 0.3)$. All curves collapse onto a single universal function $\mathcal{S}_{\alpha_1,\alpha_2}$, verifying a universal scale-invariant structure consistent with the predicted scaling law. A sharp peak emerges near ξ_{crit} consistent with the double descent phenomenon at the interpolation threshold.

Double Descent Peak. In the underparameterized limit, the loss scales as $\ell \sim N^{-(\alpha_1+\alpha_2+1)}$, while in the overparameterized limit it scales as $\ell \sim D^{-(\alpha_1+\alpha_2+1)/(\alpha_1+1)}$. This suggests a crossover at $N \sim D^{1/(\alpha_1+1)}$, which coincides with the point at which the number of model parameters N matches the number of activated (and hence learnable) coordinates $K(D)$ from Lemma 3.2. The shape of $\mathcal{S}_{\alpha_1,\alpha_2}(\xi)$ in Figure 2 reveals a prominent *double descent* phenomenon near:

$$\xi_{\text{crit}} = \frac{1}{\Gamma\left(\frac{\alpha_1}{1+\alpha_1}\right)}, \tag{9}$$

corresponding to the threshold where $K(D)$ matches N, with location determined universally by the sparsity exponent α_1.

5 Compute-Optimal Scaling Laws

We now analyze how to allocate data and model size under a fixed compute budget so as to minimize the loss. We adopt the compute model $C = ND \cdot \min\{N, D\}$ as a unified proxy for the cost of training to completion in our setting; see Appendix D for discussion. We then minimize $\ell(N, D) \sim N^{-\alpha_N} + D^{-\alpha_D}$ subject to fixed compute C to derive the compute-optimal scaling laws.

Proposition 5.1 (Compute-Optimal Frontier)**.** *Let the Bayes-optimal loss, for $\alpha_1 > 0$, scale as $\ell(N, D) \sim N^{-\alpha_N} + D^{-\alpha_D}$, with $\alpha_N = \alpha_1 + \alpha_2 + 1$ and $\alpha_D = \frac{\alpha_1+\alpha_2+1}{\alpha_1+1}$. Under a fixed compute budget $C = ND \cdot \min\{N, D\}$, the unique compute-optimal allocation lies in the underparameterized regime $N < D$, and the optimal allocation and resulting loss scale as*

$$\begin{aligned} &\ell^*(C) \sim C^{-\alpha_C}, \\ &N^*(C) \sim C^{\alpha_D/(\alpha_N+2\alpha_D)} = C^{1/(\alpha_1+3)}, \qquad D^*(C) \sim C^{1-2\alpha_D/(\alpha_N+2\alpha_D)} = C^{(\alpha_1+1)/(\alpha_1+3)}, \\ &\alpha_C := \frac{\alpha_N \alpha_D}{\alpha_N + 2\alpha_D} = \frac{\alpha_1 + \alpha_2 + 1}{\alpha_1 + 3} > 0. \end{aligned} \tag{10}$$

[8] The double-descent peak is a feature of benign interpolation under ridgeless fitting, rather than of the scaling-law picture itself. Regularization or early stopping smooths it out [12].

Remark 5.2 (Absence of Overparameterized Optimum). The compute-optimal solution always lies in the underparameterized regime: $N^*(C) < D^*(C)$ for all C. An overparameterized optimum $N > D$, with $C = ND^2$, yields $N^*(C)/D^*(C) \sim C^{(\alpha_D - \alpha_N)/(2\alpha_N + \alpha_D)} \ll 1$, contradicting the assumed regime $N > D$. Hence the only valid optimum satisfies $N < D$.

Remark 5.3 (Impact of Sparsity on Optimal Scaling). As α_1 increases, individual samples become more sparse, carrying less information per example. At the same time, increasing α_1 also steepens the marginal variance spectrum $\mathrm{Var}(x_j) = j^{-\alpha_1 - \alpha_2 - 2}$, so the residual tail loss decays more rapidly once coordinates are resolved. Consequently, for fixed $\alpha_2 < 2$, the compute-optimal exponent $\alpha_C = \frac{\alpha_1 + \alpha_2 + 1}{\alpha_1 + 3}$ increases monotonically with α_1, approaching 1 as $\alpha_1 \to \infty$. As α_1 increases, the optimal model size, $N^*(C)$, grows more slowly with compute, while the optimal dataset size, $D^*(C)$, grows more rapidly. As a result, the allocation shifts toward spending a larger fraction of compute on data rather than parameters.

Empirical Validation. To validate Proposition 5.1, Figure 3 shows test loss versus total compute C for different N. Each curve corresponds to a fixed N, while C is varied by sweeping D. The dashed line shows the predicted compute-optimal frontier, $\ell^*(C) \sim C^{-\alpha_C}$. As C increases, the empirical curves approach this frontier, confirming the predicted scaling.

6 Gradient Descent Training Dynamics

We next study the training dynamics of the readout weights $\boldsymbol{\theta}$ under full-batch GD, a broadly meaningful computational model with some generality beyond the toy setting, on the empirical squared loss $\ell(\boldsymbol{\theta}) = \frac{1}{2D}\left\|\mathbf{w}^\top \mathbf{X} - \boldsymbol{\theta}^\top \mathbf{u}\mathbf{X}\right\|_2^2$. With step size η, one GD step is $\Delta\boldsymbol{\theta} = \frac{\eta}{D}\,\mathbf{u}\mathbf{X}\mathbf{X}^\top\left(\mathbf{w} - \mathbf{u}^\top\boldsymbol{\theta}\right)$. It is convenient to rewrite the dynamics in terms of the input-space residual $\mathbf{r}_t := \mathbf{w} - \hat{\mathbf{w}}_t$, where $\hat{\mathbf{w}}_t := \mathbf{u}^\top\boldsymbol{\theta}_t \in \mathbb{R}^M$. Then

$$\mathbf{r}_{t+1} = \left(\mathbf{I}_M - \frac{\eta}{D}\,\mathbf{u}^\top\mathbf{u}\,\mathbf{X}\mathbf{X}^\top\right)\mathbf{r}_t, \quad (11)$$

so convergence is controlled by the spectrum of $\mathbf{u}^\top\mathbf{u}\,\mathbf{X}\mathbf{X}^\top$[9]; see Appendix E.

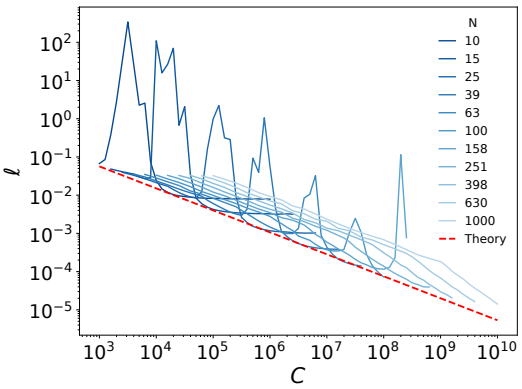

Figure 3: **Empirical scaling of loss with compute.** Each curve corresponds to test loss ℓ versus total compute C for a fixed model size N, with C varied by sweeping D, for fixed $(\alpha_1 = 1.0, \alpha_2 = 0.3)$. The dashed line denotes the predicted compute-optimal scaling $\ell^*(C) \sim C^{-\alpha_C}$. The empirical loss converges toward this frontier at high compute, just past the double descent spike for each curve demonstrating agreement between theory and experiment.

6.1 Convergence with High Probability

A first question is how η must scale for (11) to converge. Since $\ell(\boldsymbol{\theta})$ is quadratic, full-batch GD is a linear iteration, and a standard stability criterion implies that (11) converges whenever $0 < \eta < \frac{2}{\lambda_{\max}\left(\frac{1}{D}\,\mathbf{u}^\top\mathbf{u}\,\mathbf{X}\mathbf{X}^\top\right)}$, where $\lambda_{\max}(\cdot)$ denotes the largest eigenvalue [27]. Equivalently, this is the usual condition $0 < \eta < 2/\lambda_{\max}(\nabla^2_{\boldsymbol{\theta}}\ell)$, since $\nabla^2_{\boldsymbol{\theta}}\ell(\boldsymbol{\theta}) = \frac{1}{D}\mathbf{u}\mathbf{X}\mathbf{X}^\top\mathbf{u}^\top$. In the underparameterized regime, a standard concentration argument for the feature-space Hessian $\nabla^2_{\boldsymbol{\theta}}\ell(\boldsymbol{\theta}) = \frac{1}{D}\,\boldsymbol{\Phi}\boldsymbol{\Phi}^\top$ yields $\lambda_{\max}(\nabla^2_{\boldsymbol{\theta}}\ell) = 1 + o_{\mathbb{P}}(1)$ and hence any fixed $\eta \in (0, 2)$ is admissible with high probability. The overparameterized case, however, is more subtle and is controlled by whether the random embedding acts isometrically on the data span.

Proposition 6.1 (Step-size Stability with High Probability)**.** *Assume the sparse activation model of Section 2.1 with $\alpha_1 \geq -1$ and $\alpha_1 + \alpha_2 + 1 > 0$. Assume further that the random feature map is sufficiently wide that it is nearly isometric on the (low-dimensional) span of the data.*[10] *Then*

[9] The data dependence enters through the empirical covariance $\mathbf{X}\mathbf{X}^\top$. For dense inputs, $\mathrm{rank}(\mathbf{X}\mathbf{X}^\top) = \mathrm{rank}(\mathbf{X}) \approx \min\{M, D\}$, which is typically $\approx D$ when $D \lesssim M$, whereas under our sparse model, $\mathrm{rank}(\mathbf{X}\mathbf{X}^\top) \leq K(D) \asymp D^{1/(\alpha_1+1)}$, so it grows only sublinearly in D.

[10] Concretely, with $\mathbf{P}$ the orthogonal projector onto $\mathrm{span}(\mathbf{X})$, the hypothesis is $\mathbf{P}\mathbf{u}^\top\mathbf{u}\mathbf{P} = \mathbf{P} + o_{\mathbb{P}}(1)$ in operator norm. Since $\mathrm{rank}(\mathbf{X}\mathbf{X}^\top) \leq K(D)$, standard random-projection bounds give this for $N \gg K(D)$, with no dependence on M; see Appendix E.1.

$$\lambda_{\max}\left(\frac{1}{D}\,\mathbf{u}^\top\mathbf{u}\,\mathbf{X}\mathbf{X}^\top\right) = 1 + o_{\mathbb{P}}(1), \tag{12}$$

and consequently, for any fixed $\eta \in (0, 2)$, the iteration (11) *is stable and converges with high probability.*

6.2 A Scaling Law for Failure of Gradient Descent

Proposition 6.1 is a high-probability statement; rare datasets can still contain a single rare activation spike that produces an anomalously large top eigenvalue and causes divergence.[11] GD becomes unstable when $\lambda_{\max}\big(\frac{\eta}{D}\mathbf{u}^\top\mathbf{u}\mathbf{X}\mathbf{X}^\top\big) > 2$. Under the near-isometry approximation $\mathbf{u}^\top\mathbf{u} \approx \mathbf{I}_M$ on the data span, this reduces to $\lambda_{\max}\big(\mathbf{X}\mathbf{X}^\top\big) > \frac{2D}{\eta}$. This is only a concern when activated amplitudes *grow* with index, *i.e.* when $\alpha_2 < -1$ (since then $x_j^2 = j^{-(\alpha_2+1)}$ increases with j on activation).

Theorem 6.2 (Failure Probability of GD)**.** *Fix $\eta \in (0, 2)$ and suppose $\alpha_2 < -1$. Under the sparse activation model of Section 2.1, with $\alpha_1 \geq -1$ and $\alpha_1 + \alpha_2 + 1 > 0$, and under the near-isometry hypothesis of Proposition 6.1, let $\nu := \frac{\alpha_1+\alpha_2+1}{-\alpha_2-1} > 0$. Then the probability that full-batch GD is unstable due to a rare spike on a random dataset of size D obeys*

$$\mathbb{P}_{\text{rare}}(\text{diverge}) \asymp D^{-\nu}, \tag{13}$$

up to η-dependent constants.

This theorem identifies instability under a fixed step size, not a fundamental optimization barrier. Reducing the learning rate or using clipped updates can restore stability [30].

7 Experiments with Nonlinear Activations

A natural question is whether the two-exponent asymmetry we identify is specific to the linear setting or reflects a more general property of sparse data distributions. In Figure 4, we test robustness to nonlinearity by training a two-layer network with hidden layer $\boldsymbol{\phi} = \sigma(\mathbf{u}\mathbf{x})$ (ReLU activation, frozen random first-layer weights) and a linear readout, using full-batch GD with Nesterov momentum and adaptive restart; see Appendix F for experimental details. Our dense baseline is Gaussian with the same per-coordinate variance $j^{-(\alpha_1+\alpha_2+2)}$ as the sparse model, so the two designs share their covariance and differ only in whether every coordinate is active in every sample. The fitted exponents shift downward in the model-limited regime, consistent with the general expectation that nonlinear feature maps smooth power-law input spectra [25, 31, 32]. Crucially, the qualitative asymmetry persists: sparse N-, dense N-, and dense D-sweeps are well-described by a single shared exponent, while the sparse D-sweep is better described by a distinct exponent. This supports our main result: the bottleneck mechanism—unobserved coordinates carrying no signal—is robust to nonlinearity and first-order optimization.

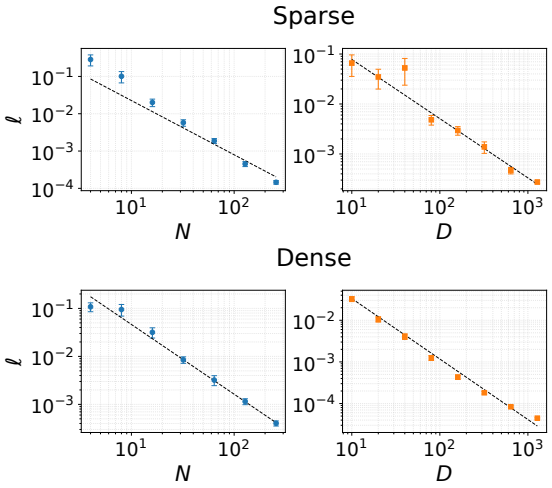

Figure 4: **Asymmetry persists under nonlinearity.** Test loss scaling under the ReLU feature map $\boldsymbol{\phi} = \sigma(\mathbf{u}\mathbf{x})$, trained with Nesterov + adaptive restart; $(\alpha_1, \alpha_2) = (1.0, 0.3)$, 5 seeds. Top: sparse; bottom: dense. Left: N-sweep at $D = 50{,}000$; right: D-sweep at $N = 8000$. A single exponent $\alpha \approx 1.5$ (dashed lines, jointly fitted with separate intercepts) describes the sparse N-, dense N-, and dense D-sweeps. The sparse D-sweep requires a shallower fit, $\alpha_D \approx 1.2$, breaking the dense-baseline symmetry $\alpha_N = \alpha_D$ predicted by linear theory.

8 Experiments on Pile Token Statistics

We investigate the influence of sparsity on scaling in a more realistic setting. We consider the Pile dataset [16], where we take each sample to be a single token, with the active coordinate j drawn

[11] Rare activations can create spectral outliers in $\mathbf{X}\mathbf{X}^\top$ that are not controlled by the high-probability spectral concentration arguments used in recent approximation/bias/variance analyses of linear-regression scaling laws [18, 28, 29], so we instead adopt a probabilistic treatment.

from the measured Zipfian token frequencies p_j. We compare against a dense baseline with the same per-coordinate variances, in which every coordinate is active in every sample. In both cases we take the amplitude exponent $\alpha_2 = 0.3$ and a flat target (w_j i.i.d., unit variance). See Appendix G for experimental details. For linear random features at fixed N, the double-descent peak of the dense baseline falls at $D = N$, as expected, while that of the sparse design falls at $K(D) = N$ (Figure 5), with $K(D)$ the number of distinct tokens counted among D samples, consistent with our analysis in Figure 2. Fitting both exponents over the same range of effective rank, 90 to 4300, with the effective rank N in the N-sweep and $K(D)$ (sparse) or D (dense) in the D-sweep, and with p_j measured over 1.48×10^8 Pile tokens with 147,775 distinct types, the sparse design gives $\alpha_N - \alpha_D = +0.062\,[+0.021, +0.101]$ and the dense baseline $-0.020\,[-0.060, +0.020]$, a difference of $+0.082\,[+0.026, +0.133]$, showing that the asymmetry grows with sparsity. The gap is small, comparable to the difference between the parameter and data exponents, 0.34 and 0.28, reported in Hoffmann et al. [2]. Finally, exploratory experiments on a depth-3 ReLU network of width H acting on an N-dimensional random projection and trained in the lazy regime (Appendix G) confirm that the loss tracks the variance carried by unobserved tokens as D grows, independently of H.

9 Conclusion

We introduced a model that gives rise to two distinct scaling exponents under sparse input activations. By analyzing the population loss in both under- and overparameterized regimes, we showed how sparsity breaks the symmetry of scaling laws in the random feature setting. We derived a compute-optimal scaling law showing that sparsity enhances efficiency by accelerating the decay of test loss with compute, with a frontier that prioritizes data over model size. Analyzing GD dynamics, we proved high-probability stability for all step sizes below 2 and identified a suppressed sublinear scaling law for its probability of failure in the overparameterized regime, suggesting a role for gradient clipping in practical optimization. We validated these results experimentally and showed that the asymmetry is robust to nonlinearity. We further confirmed fingerprints of this behavior on language data from the Pile dataset. Our main takeaway is that sparsity as a structural property fundamentally shapes scaling behavior in high-dimensional learning.

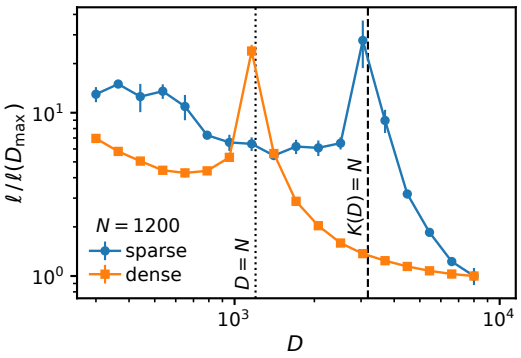

Figure 5: **Double-descent peak on Pile token statistics.** Test loss ℓ of linear random features with $N = 1200$ as a function of D, normalized to its value at the largest D, $D_{\max}$, for the sparse design and the dense baseline (4 seeds, ± 1 standard error). The peak corresponds to the interpolation threshold, where the rank of $\mathbf{X}\mathbf{X}^\top$ reaches N. In the dense baseline, the peak falls at $D = N$ (dotted). In the sparse design, the rank is $K(D) < D$, so D must grow past N before $K(D)$ reaches N, and the peak moves right, to $K(D) = N$ (dashed).

Limitations. Our results are built around data sparsity as the organizing principle, yet several extensions remain for future work. Incorporating exogenous label noise, which we expect to impose an error floor rather than remove the sparsity-driven bottleneck until it dominates the optimization scale, and analyzing ridge regularization would extend our picture to noisy and explicitly regularized settings. More ambitiously, a theoretical treatment of nonlinear feature maps—complementing our empirical validation—and an extension to the dynamical regime of few- to multi-pass SGD both confront a common technical obstacle: the non-Gaussian, heavy-tailed nature of our features requires extending existing deterministic-equivalent tools beyond the Gaussian setting [14, 22]. Relatedly, extending our analysis to the feature-learning regime is an important direction. We do not expect feature learning by itself to remove the coverage bottleneck, since its advantage over a fixed feature map appears to lie in adapting to low-dimensional structure in the target [33], whereas here sparsity is in which coordinates are observed rather than in which carry signal. Feature learning could, however, relax the independence assumption, and correlated features might route signal to rarely observed coordinates, or concentrate capacity on them [19]. Conversely, feature learning may also induce sparse representations of its own, which hurt generalization in smooth tasks [34]. A further natural direction is to test these effects at scale, in deeper networks and in transformers, particularly in light of sparse-feature observations from sparse autoencoders [35, 36] and recent connections between superposition [37] and scaling laws [38].

Acknowledgments and Disclosure of Funding

We thank E. Paquette for helpful discussions and feedback on an earlier version of the manuscript. This research was supported in part by the Yale Office of the Provost. The authors declare no competing interests.

References

[1] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[2] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models. In *Advances in Neural Information Processing Systems (NeurIPS 2022)*, volume 35, pages 30016–30030, 2022.

[3] Blake Bordelon, Abdulkadir Canatar, and Cengiz Pehlevan. Spectrum dependent learning curves in kernel regression and wide neural networks. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1024–1034, 2020.

[4] Stefano Spigler, Mario Geiger, and Matthieu Wyart. Asymptotic learning curves of kernel methods: Empirical data versus teacher–student paradigm. *Journal of Statistical Mechanics: Theory and Experiment*, 2020(12):124001, 2020. doi: 10.1088/1742-5468/abc61d.

[5] Yasaman Bahri, Ethan Dyer, Jared Kaplan, Jaehoon Lee, and Utkarsh Sharma. Explaining neural scaling laws. *arXiv preprint arXiv:2102.06701*, 2021.

[6] Andrew Maloney, Daniel A. Roberts, and James Sully. A solvable model of neural scaling laws. *arXiv preprint arXiv:2210.16859*, 2022.

[7] Blake Bordelon, Alexander Atanasov, and Cengiz Pehlevan. A dynamical model of neural scaling laws. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 4345–4382, 2024.

[8] Elliot Paquette, Courtney Paquette, Lechao Xiao, and Jeffrey Pennington. 4+3 phases of compute-optimal neural scaling laws. In *Advances in Neural Information Processing Systems (NeurIPS 2024)*, volume 37, pages 16459–16537, 2024.

[9] Mario Geiger, Stefano Spigler, Stéphane d'Ascoli, Levent Sagun, Marco Baity-Jesi, Giulio Biroli, and Matthieu Wyart. Jamming transition as a paradigm to understand the loss landscape of deep neural networks. *Physical Review E*, 100(1):012115, 2019. doi: 10.1103/PhysRevE.100.012115.

[10] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019. doi: 10.1073/pnas.1903070116.

[11] Mikhail Belkin, Daniel Hsu, and Ji Xu. Two models of double descent for weak features. *SIAM Journal on Mathematics of Data Science*, 2(4):1167–1180, 2020. doi: 10.1137/20M1336072.

[12] Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J. Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *The Annals of Statistics*, 50(2):949–986, 2022. doi: 10.1214/21-AOS2133.

[13] Peter L. Bartlett, Philip M. Long, Gábor Lugosi, and Alexander Tsigler. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 117(48):30063–30070, 2020. doi: 10.1073/pnas.1907378117.

[14] Song Mei and Andrea Montanari. The generalization error of random features regression: Precise asymptotics and the double descent curve. *Communications on Pure and Applied Mathematics*, 75(4):667–766, 2022. doi: 10.1002/cpa.22008.

[15] Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. In *International Conference on Learning Representations (ICLR)*, 2020. doi: 10.48550/arXiv.1912.02292.

[16] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The Pile: An 800GB dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

[17] Leonardo Defilippis, Bruno Loureiro, and Theodor Misiakiewicz. Dimension-free deterministic equivalents and scaling laws for random feature regression. In *Advances in Neural Information Processing Systems (NeurIPS 2024)*, volume 37, pages 104630–104693, 2024.

[18] Licong Lin, Jingfeng Wu, Sham M. Kakade, Peter L. Bartlett, and Jason D. Lee. Scaling laws in linear regression: Compute, parameters, and data. In *Advances in Neural Information Processing Systems (NeurIPS 2024)*, volume 37, pages 60556–60606, 2024.

[19] Blake Bordelon, Alexander Atanasov, and Cengiz Pehlevan. How feature learning can improve neural scaling laws. *Journal of Statistical Mechanics: Theory and Experiment*, 2025(8):084002, 2025. doi: 10.1088/1742-5468/adefb1.

[20] Federica Gerace, Bruno Loureiro, Florent Krzakala, Marc Mézard, and Lenka Zdeborová. Generalisation error in learning with random features and the hidden manifold model. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 3452–3462, 2020.

[21] Sebastian Goldt, Marc Mézard, Florent Krzakala, and Lenka Zdeborová. Modelling the influence of data structure on learning in neural networks: the hidden manifold model. *Physical Review X*, 10(4):041044, 2020. doi: 10.1103/PhysRevX.10.041044.

[22] Hong Hu and Yue M. Lu. Universality laws for high-dimensional learning with random features. *IEEE Transactions on Information Theory*, 69(3):1932–1964, 2023. doi: 10.1109/TIT.2022.3217698.

[23] Sebastian Goldt, Bruno Loureiro, Galen Reeves, Florent Krzakala, Marc Mézard, and Lenka Zdeborová. The Gaussian equivalence of generative models for learning with shallow neural networks. In *Proceedings of the 2nd Mathematical and Scientific Machine Learning Conference*, volume 145 of *Proceedings of Machine Learning Research*, pages 426–471, 2022.

[24] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems (NeurIPS 2007)*, volume 20, pages 1177–1184, 2007.

[25] Youngmin Cho and Lawrence K. Saul. Kernel methods for deep learning. In *Advances in Neural Information Processing Systems (NeurIPS 2009)*, volume 22, pages 342–350, 2009.

[26] Zhengkang Zhang. Neural scaling laws from large-N field theory: Solvable model beyond the ridgeless limit. *arXiv preprint arXiv:2405.19398*, 2024.

[27] Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Kluwer Academic Publishers, 2004.

[28] Licong Lin, Jingfeng Wu, and Peter L. Bartlett. Improved scaling laws in linear regression via data reuse. In *Advances in Neural Information Processing Systems (NeurIPS 2025)*, 2025.

[29] Tingkai Yan, Haodong Wen, Binghui Li, Kairong Luo, Wenguang Chen, and Kaifeng Lyu. Larger datasets can be repeated more: A theoretical analysis of multi-epoch scaling in linear regression. In *The Fourteenth International Conference on Learning Representations*, 2026.

[30] Ta Duy Nguyen, Thien H. Nguyen, Alina Ene, and Huy Nguyen. Improved convergence in high probability of clipped gradient methods with heavy tailed noise. In *Advances in Neural Information Processing Systems (NeurIPS 2023)*, volume 36, pages 24191–24222, 2023.

[31] Cosme Louart, Zhenyu Liao, and Romain Couillet. A random matrix approach to neural networks. *The Annals of Applied Probability*, 28(2):1190–1248, 2018. doi: 10.1214/17-AAP1328.

[32] Alberto Bietti and Julien Mairal. On the inductive bias of neural tangent kernels. In *Advances in Neural Information Processing Systems (NeurIPS 2019)*, volume 32, pages 12873–12884, 2019.

[33] Behrooz Ghorbani, Song Mei, Theodor Misiakiewicz, and Andrea Montanari. When do neural networks outperform kernel methods? In *Advances in Neural Information Processing Systems (NeurIPS 2020)*, volume 33, pages 14820–14830, 2020.

[34] Leonardo Petrini, Francesco Cagnetta, Eric Vanden-Eijnden, and Matthieu Wyart. Learning sparse features can lead to overfitting in neural networks. In *Advances in Neural Information Processing Systems (NeurIPS 2022)*, volume 35, pages 9403–9416, 2022.

[35] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E. Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023.

[36] Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Adam Jermyn, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E. Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Scaling monosemanticity: Extracting interpretable features from Claude 3 Sonnet. *Transformer Circuits Thread*, 2024.

[37] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition. *Transformer Circuits Thread*, 2022.

[38] Yizhou Liu, Ziming Liu, and Jeff Gore. Superposition yields robust neural scaling. In *Advances in Neural Information Processing Systems (NeurIPS 2025)*, 2025.

[39] Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo Jun, Tom B. Brown, Prafulla Dhariwal, Scott Gray, Chris Hallacy, Benjamin Mann, Alec Radford, Aditya Ramesh, Nick Ryder, Daniel M. Ziegler, John Schulman, Dario Amodei, and Sam McCandlish. Scaling laws for autoregressive generative modeling. *arXiv preprint arXiv:2010.14701*, 2020.

[40] Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md. Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017.

[41] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12104–12113, 2022.

[42] Tomer Porian, Mitchell Wortsman, Jenia Jitsev, Ludwig Schmidt, and Yair Carmon. Resolving discrepancies in compute-optimal scaling of language models. In *Advances in Neural Information Processing Systems (NeurIPS 2024)*, volume 37, 2024.

[43] Rylan Schaeffer, Noam Levi, Andreas Kirsch, Theo Guenais, Brando Miranda, Elyas Obbad, and Sanmi Koyejo. Evaluating the robustness of Chinchilla compute-optimal scaling. *arXiv preprint arXiv:2509.23963*, 2025.

[44] Shane Bergsma, Nolan Dey, Gurpreet Gosal, Gavia Gray, Daria Soboleva, and Joel Hestness. Power lines: Scaling laws for weight decay and batch size in LLM pre-training. In *Advances in Neural Information Processing Systems (NeurIPS 2025)*, 2025.

[45] Tanishq Kumar, Zachary Ankner, Benjamin Frederick Spector, Blake Bordelon, Niklas Muennighoff, Mansheej Paul, Cengiz Pehlevan, Christopher Ré, and Aditi Raghunathan. Scaling laws for precision. In *The Thirteenth International Conference on Learning Representations*, 2025.

[46] Nikhil Sardana, Jacob Portes, Sasha Doubov, and Jonathan Frankle. Beyond Chinchilla-optimal: Accounting for inference in language model scaling laws. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 43445–43460, 2024.

[47] Peihao Wang, Rameswar Panda, and Zhangyang Wang. Data efficient neural scaling law via model reusing. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 36193–36204, 2023.

[48] Niklas Muennighoff, Alexander M. Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin A. Raffel. Scaling data-constrained language models. In *Advances in Neural Information Processing Systems (NeurIPS 2023)*, volume 36, pages 50358–50376, 2023.

[49] Jean Barbier, Florent Krzakala, Nicolas Macris, Léo Miolane, and Lenka Zdeborová. Optimal errors and phase transitions in high-dimensional generalized linear models. *Proceedings of the National Academy of Sciences*, 116(12):5451–5460, 2019. doi: 10.1073/pnas.1802705116.

[50] Benjamin Aubin, Antoine Maillard, Jean Barbier, Florent Krzakala, Nicolas Macris, and Lenka Zdeborová. The committee machine: Computational to statistical gaps in learning a two-layers neural network. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124023, 2019. doi: 10.1088/1742-5468/ab43d2.

[51] Song Mei, Theodor Misiakiewicz, and Andrea Montanari. Learning with invariances in random features and kernel models. In *Proceedings of the Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pages 3351–3418, 2021.

[52] Song Mei, Theodor Misiakiewicz, and Andrea Montanari. Generalization error of random feature and kernel methods: Hypercontractivity and kernel matrix concentration. *Applied and Computational Harmonic Analysis*, 59:3–84, 2022. doi: 10.1016/j.acha.2021.12.003.

[53] Hugo Latourelle-Vigeant and Elliot Paquette. Dyson equation for correlated linearizations and test error of random features regression. *arXiv preprint arXiv:2312.09194*, 2023.

[54] Alexander Atanasov, Jacob A. Zavatone-Veth, and Cengiz Pehlevan. Scaling and renormalization in high-dimensional regression. *arXiv preprint arXiv:2405.00592*, 2024.

[55] Arie Wortsman and Bruno Loureiro. Kernel ridge regression under power-law data: Spectrum and generalization. *arXiv preprint arXiv:2510.04780*, 2025.

[56] Elliot Paquette, Ke Liang Xiao, and Yizhe Zhu. Power-law spectrum of the random feature model. *arXiv preprint arXiv:2603.14578*, 2026.

[57] Jean Barbier, Francesco Camilli, Minh-Toan Nguyen, Mauro Pastore, and Rudy Skerk. Statistical physics of deep learning: Optimal learning of a multi-layer perceptron near interpolation. *arXiv preprint arXiv:2510.24616*, 2025.

[58] Leonardo Defilippis, Yizhou Xu, Julius Girardin, Emanuele Troiani, Vittorio Erba, Lenka Zdeborová, Bruno Loureiro, and Florent Krzakala. Scaling laws and spectra of shallow neural networks in the feature learning regime. In *The Fourteenth International Conference on Learning Representations*, 2026.

[59] Dechen Zhang, Xuan Tang, Yingyu Liang, and Difan Zou. Scaling laws for precision in high-dimensional linear regression. *arXiv preprint arXiv:2602.19241*, 2026.

[60] Francesco Cagnetta, Allan Raventós, Surya Ganguli, and Matthieu Wyart. Deriving neural scaling laws from the statistics of natural language. *arXiv preprint arXiv:2602.07488*, 2026.

[61] Maissam Barkeshli, Alberto Alfarano, and Andrey Gromov. On the origin of neural scaling laws: From random graphs to natural language. *arXiv preprint arXiv:2601.10684*, 2026.

[62] Binghui Li, Fengling Chen, Zixun Huang, Lean Wang, and Lei Wu. Functional scaling laws in kernel regression: Loss dynamics and learning rate schedules. In *Advances in Neural Information Processing Systems (NeurIPS 2025)*, 2025.

[63] Blake Bordelon and Francesco Mori. Theory of optimal learning rate schedules and scaling laws for a random feature model. *arXiv preprint arXiv:2602.04774*, 2026.

[64] Jakob Kramp, Javed Lindner, and Moritz Helias. Dynamics of neural scaling laws in random feature regression with powerlaw-distributed kernel eigenvalues. *arXiv preprint arXiv:2602.23039*, 2026.

[65] Yunwei Ren, Eshaan Nichani, Denny Wu, and Jason D. Lee. Emergence and scaling laws in SGD learning of shallow neural networks. In *Advances in Neural Information Processing Systems (NeurIPS 2025)*, 2025.

[66] Gérard Ben Arous, Murat A. Erdogdu, N. Mert Vural, and Denny Wu. Learning quadratic neural networks in high dimensions: SGD dynamics and scaling laws. In *Advances in Neural Information Processing Systems (NeurIPS 2025)*, 2025.

[67] Roman Worschech and Bernd Rosenow. Analyzing neural scaling laws in two-layer networks with power-law data spectra. In *The Thirteenth International Conference on Learning Representations*, 2025.

[68] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 15 of *Proceedings of Machine Learning Research*, pages 315–323, 2011.

[69] Bruno A. Olshausen and David J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, 1996. doi: 10.1038/381607a0.

[70] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *The Fifth International Conference on Learning Representations*, 2017.

[71] Jan Ludziejewski, Jakub Krajewski, Kamil Adamczewski, Maciej Pióro, Michał Krutul, Szymon Antoniak, Kamil Ciebiera, Krystian Król, Tomasz Odrzygóźdź, Piotr Sankowski, Marek Cygan, and Sebastian Jaszczur. Scaling laws for fine-grained mixture of experts. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 33270–33288, 2024.

[72] Samira Abnar, Harshay Shah, Dan Busbridge, Alaaeldin El-Nouby, Joshua M. Susskind, and Vimal Thilak. Parameters vs FLOPs: Scaling laws for optimal sparsity for mixture-of-experts language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 204–230, 2025.

[73] Yurii Nesterov. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. *Doklady Akademii Nauk SSSR*, 269:543–547, 1983.

[74] Brendan O'Donoghue and Emmanuel Candès. Adaptive restart for accelerated gradient schemes. *Foundations of Computational Mathematics*, 15(3):715–732, 2015. doi: 10.1007/s10208-013-9150-3.

[75] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems (NeurIPS 2018)*, volume 31, pages 8571–8580, 2018.

A Additional Related Work

Empirical scaling laws relating test loss to parameters, data, and compute were systematically documented in language modeling [1] and refined into compute-optimal prescriptions emphasizing data-limited training in LLMs [2]. Related scaling behavior has also been observed across tasks and modalities, including autoregressive generative modeling [39], multi-domain studies [40] and vision transformer scaling analyses [41]. More recent empirical work has expanded scaling-law analyses to address compute-optimal discrepancies across experimental protocols [42, 43], practical scaling laws for weight decay and batch size in LLM pretraining [44], precision-aware scaling under quantization and training precision [45], inference-aware compute-optimal scaling [46], data-scarcity regimes via model reuse [47], and data-constrained scaling under token repetition [48].

On the theory side, a growing literature develops tractable models that reproduce and explain scaling-law phenomenology. Statistical-physics teacher-student models characterize optimal errors and phase transitions in high-dimensional generalized linear models [49] and computational-to-statistical gaps in committee machines [50]. Solvable neural-scaling models connect power-law learning curves to spectral structure and compute-optimal frontiers [5, 6]. Random-feature and kernel analyses address invariances [51], hypercontractivity and kernel matrix concentration [52], correlated linearizations and test error [53], and dimension-free deterministic equivalents [17]. Scaling and renormalization approaches provide a complementary high-dimensional regression perspective [54]. Recent work also extends this power-law spectral perspective to kernel ridge regression under anisotropic power-law data [55] and spectral inheritance through random-feature maps [56]. Related statistical-physics analyses study feature learning and specialization in finite-width teacher-student neural networks near interpolation [57], as well as scaling-law phase diagrams in shallow feature-learning networks [58]. A complementary line studies compute-optimal scaling in linear regression [18] and precision-aware variants in high-dimensional linear regression [59]. Recent work also studies the origin of scaling laws in simplified language and sequence-modeling settings, including theories based on natural-language statistics [60] and controlled graph/random-walk generative processes [61].

Beyond Bayes-optimal analyses, several works study *training dynamics*. Dynamical models of one-pass or finite-time training exhibit multiple scaling phases and shift compute-optimal allocations [7, 8]. Feature learning can also change scaling exponents and compute-optimal behavior [19]. Data reuse and multi-epoch training shift effective exponents and optimal allocations [28, 29]. Functional loss dynamics and learning-rate schedules in kernel or random-feature regression are studied in [62, 63], while early stopping in random-feature regression with power-law spectra is analyzed in [64]. Related analyses of SGD in shallow or two-layer networks with heterogeneous teacher components [65], quadratic feature-learning models [66], or power-law data spectra [67] derive sample-, time-, and parameter-dependent scaling exponents.

Our work is motivated by a different axis: *sparse or conditional feature activation* [68, 69], in which only a subset of features is observed or used for any example. Such sparsity is common in modern representation learning and conditional computation (*e.g.*, mixture-of-experts) [70]. Related empirical work studies scaling laws for sparse conditional computation in mixture-of-experts models [71, 72]. Rather than refining existing spectral-decay accounts, we isolate how sparsity in what the dataset reveals can itself produce an intrinsic asymmetry between model-limited and data-limited scaling exponents.

B Population Objective of Sparse Model

We begin by reformulating the objective in Eq. (2) as the following loss function:

$$\begin{aligned}
\ell &= \frac{\gamma}{D}\|\boldsymbol{\theta}\|_2^2 + \frac{1}{D}\sum_{a=1}^{D}\Big(y(\mathbf{x}_a) - \boldsymbol{\theta}^\top \mathbf{u}\mathbf{x}_a\Big)^2 \\
&= \frac{\gamma}{D}\|\boldsymbol{\theta}\|_2^2 + \frac{1}{D}\sum_{a=1}^{D}\Big(\mathbf{w}^\top \mathbf{x}_a - \boldsymbol{\theta}^\top \mathbf{u}\mathbf{x}_a\Big)^2 \\
&= \frac{\gamma}{D}\|\boldsymbol{\theta}\|_2^2 + \frac{1}{D}\sum_{a=1}^{D}\Big[\mathbf{w}^\top \mathbf{x}_a\mathbf{x}_a^\top \mathbf{w} - 2\,\boldsymbol{\theta}^\top \mathbf{u}\mathbf{x}_a\mathbf{x}_a^\top \mathbf{w} + \boldsymbol{\theta}^\top \mathbf{u}\mathbf{x}_a\mathbf{x}_a^\top \mathbf{u}^\top \boldsymbol{\theta}\Big],
\end{aligned} \tag{14}$$

which we wish to minimize with respect to $\boldsymbol{\theta}$. While our focus is on the ridge-less limit, $\gamma = 0$, we retain it for mathematical convenience and take the limit $\gamma \to 0$ when appropriate.

Using $\boldsymbol{\Phi} = \mathbf{u}\mathbf{X}$, we find the optimal solution $\boldsymbol{\theta}^\star$ satisfies:

$$\boldsymbol{\theta}^{\star\top} = \mathbf{y}\boldsymbol{\Phi}^\top\mathbf{q} = \mathbf{y}\mathbf{Q}\boldsymbol{\Phi}^\top, \tag{15}$$

where

$$\mathbf{q} = (\gamma\,\mathbf{I}_N + \boldsymbol{\Phi}\boldsymbol{\Phi}^\top)^{-1},$$
$$\mathbf{Q} = (\gamma\,\mathbf{I}_D + \boldsymbol{\Phi}^\top\boldsymbol{\Phi})^{-1}.$$

We are interested in the difference between the effective weight vector $\mathbf{w}_{\text{eff}}$, defined by

$$\mathbf{w}_{\text{eff}}^\top = \boldsymbol{\theta}^{\star\top}\mathbf{u},$$

and the true target weights $\mathbf{w}^\top$. In particular, the generalization error is given by

$$\mathcal{E}_{\text{gen}} = \left(\mathbf{w}_{\text{eff}}^\top - \mathbf{w}^\top\right)\boldsymbol{\Sigma}_{\mathbf{x}}\left(\mathbf{w}_{\text{eff}} - \mathbf{w}\right).$$

Table 1 lists the shapes and elementwise scalings used in our computations.

Table 1: Shapes and elementwise scaling.

Parameter	Description	Shape	Scaling (elementwise)
$\mathbf{X}$	Training data (columns are examples)	$M \times D$	per-coordinate standard deviation $j^{-(\alpha_1+\alpha_2+2)/2}$
$\mathbf{u}$	Fixed random embedding	$N \times M$	$u_{ij} \sim \mathcal{N}(0, 1/N)$ i.i.d.
$\boldsymbol{\theta}$	Trainable readout	$N \times 1$	$O(1)$
$\mathbf{w}$	Target linear functional	$M \times 1$	$O(1)$
$\mathbf{y}$	Target function	$1 \times D$	$O(1)$
y	Per-example target function value	scalar	$O(1)$
$\boldsymbol{\Phi}$	Feature matrix $\mathbf{u}\mathbf{X}$	$N \times D$	N/A
$\boldsymbol{\phi}$	Features for a single input, $\mathbf{u}\mathbf{x}$	$N \times 1$	N/A
$\mathbf{Q}$	Feature Gram $(\gamma\mathbf{I}_D + \boldsymbol{\Phi}^\top\boldsymbol{\Phi})^{-1}$	$D \times D$	N/A
$\mathbf{q}$	Dual Gram $(\gamma\mathbf{I}_N + \boldsymbol{\Phi}\boldsymbol{\Phi}^\top)^{-1}$	$N \times N$	N/A

C Scaling Asymptotics of Sparse Random Features

C.1 $-1 \le \alpha_1 \le 0$: One-Exponent Scaling Law

In our sparse model of Section 2.1, we obtain a symmetric one-exponent scaling law when $-1 \le \alpha_1 \le 0$, similar to dense random feature models [5, 6]. The key point is that generalization can be bottlenecked by three resources: the number of parameters N, the number of samples D, and the set of input coordinates that are actually observed (activated at least once) in the D training examples.

Recall that the activation probability of coordinate j is

$$p_j := \mathbb{P}(x_j \neq 0) = j^{-(\alpha_1+1)}.$$

Across D i.i.d. samples, let N_j denote the number of times coordinate j activates. Then

$$\mathbb{P}(\text{coordinate } j \text{ is never active in the training set}) = \mathbb{P}(N_j = 0) = (1-p_j)^D \approx e^{-Dp_j}.$$

Thus, whenever $Dp_j \gg 1$ (equivalently, $D\,j^{-(\alpha_1+1)} \gg 1$), we have $\mathbb{P}(N_j = 0) \le e^{-\Omega(1)} \ll 1$, so coordinate j is activated at least once with high probability. When $-1 \le \alpha_1 < 0$, this condition holds uniformly for all j up to order D. At the boundary $\alpha_1 = 0$, the coordinate $j \sim D$ has $Dp_j = O(1)$, so the never-observed probability is also $O(1)$; this boundary case only affects constants and does not change the one-exponent scaling.

Therefore the Bayes-optimal loss is governed by the unmodeled-variance tail beyond the first $\min\{N, D\}$ coordinates:

$$\ell_{\text{Bayes}}(N, D) \asymp \sum_{j > \min\{N,D\}} \text{Var}(x_j) \asymp \sum_{j > \min\{N,D\}} j^{-(\alpha_1+\alpha_2+2)} \asymp \min\{N, D\}^{-(\alpha_1+\alpha_2+1)},$$

and the scaling is symmetric in N and D in this regime, with a single exponent $\alpha_N = \alpha_D = \alpha_1 + \alpha_2 + 1$.

C.2 $\alpha_1 > 0$: Two-Exponent Scaling Law

Here we briefly elaborate on supplementary details omitted from the main text regarding the scaling asymptotics of the two-exponent scaling law.

In a random set of D data points, only $O(D^{\frac{1}{1+\alpha_1}})$ features are likely to activate when $\alpha_1 > 0$. Thus, for positive α_1, there are naively three distinct regimes:

$$N < D^{\frac{1}{1+\alpha_1}}, \quad D^{\frac{1}{1+\alpha_1}} < N < D, \quad \text{and} \quad N > D.$$

The first is unambiguously underparameterized, the third is clearly overparameterized, and the intermediate regime exhibits characteristics of both.

C.2.1 Underparameterized Regime

We first consider the underparameterized regime $N \ll D^{\frac{1}{1+\alpha_1}}$, where learning is primarily limited by model capacity.

Proof of Proposition 3.1

Proof. A resolvent analysis of random-feature regression [6] reveals that the loss is governed by the covariance mass missed by an N-dimensional random feature subspace. In the ridgeless limit, the resolvent acts as a projection onto the latent directions inaccessible to the random features, so the leading-order loss is controlled by the unresolved tail of the covariance spectrum. Thus, at the level of scaling, the random-feature bottleneck resolves the leading N spectral directions and leaves the remaining covariance tail unresolved. In the present regime $N \ll D^{1/(1+\alpha_1)}$, sparsity is masked, so the relevant spectrum is $\mathrm{Var}(x_j) = j^{-\alpha_1-\alpha_2-2}$. Hence the residual variance scales as

$$\begin{aligned}\sum_{j>N} j^{-\alpha_1-\alpha_2-2} &\asymp \int_N^\infty j^{-\alpha_1-\alpha_2-2}\, dj \\ &= \frac{1}{\alpha_1+\alpha_2+1}\, N^{-(\alpha_1+\alpha_2+1)}.\end{aligned}$$

Combining this with the general form of the Bayes-optimal loss (Eq. (3)) yields (4). □

C.2.2 Overparameterized Regime

We now turn to the overparameterized regime $N \gg D^{\frac{1}{1+\alpha_1}}$.

Heavily Overparameterized Regime We first focus on the maximally overparameterized case. We will show that in this regime, the performance of the random feature model is Bayes-optimal among all models, and we compute its generalization error.

Optimal Predictor. An upper bound on the performance of any model would be for $\mathbf{w}_{\text{eff}}$ to equal $\mathbf{w}$ on indices that appear in the dataset, and be 0 on all other indices. We will show that this is indeed what happens in this regime.

If we write out our expression for $\mathbf{w}_{\text{eff}}$, we have

$$\mathbf{w}_{\text{eff}}^\top = \mathbf{w}^\top \mathbf{X} \left(\gamma \mathbf{I}_D + \mathbf{X}^\top \mathbf{u}^\top \mathbf{u}\, \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{u}^\top \mathbf{u}, \tag{16}$$

If we post-multiply by $\mathbf{X}\,\mathbf{b}$ for any vector $\mathbf{b}$, we obtain

$$\mathbf{w}_{\text{eff}}^\top \mathbf{X}\,\mathbf{b} = \mathbf{w}^\top \mathbf{X}\,\mathbf{b}.$$

This follows from the full rank of $\mathbf{Q}$. Thus, in the overparameterized regime, for every possible input in the subspace spanned by the training data, the random feature model incurs zero loss.

Coefficients Outside the Training Data Go to 0**.** While the analysis showing that $\mathbf{w}_{\text{eff}}$ perfectly interpolates the training data applies for any overparameterized model in the limit $\gamma \to 0$, the argument in this subsection uses the stronger assumption that we are deep in the overparameterized regime. Since $u_{ij} \sim \mathcal{N}(0, 1/N)$, the diagonal entries of $\mathbf{u}^\top \mathbf{u}$ are $1 + O_{\mathbb{P}}(N^{-1/2})$, while the off-diagonal entries are $O_{\mathbb{P}}(N^{-1/2})$. Thus, for any subspace S with $\dim(S) \ll N$, the restriction of $\mathbf{u}^\top \mathbf{u}$ to S is $\mathbf{I}_S + o_{\mathbb{P}}(1)$. Since the row and column spaces of $\mathbf{X}$ are at most D-dimensional, when $D \ll N$ we can replace the occurrences of $\mathbf{u}^\top \mathbf{u}$ in Eq. (16) by $\mathbf{I}$. Therefore, $\mathbf{w}_{\text{eff}}$ simplifies to

$$\mathbf{w}_{\text{eff}}^\top = \mathbf{w}^\top \mathbf{X} \left(\gamma \mathbf{I}_D + \mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top.$$

In the limit $\gamma \to 0$, this is precisely the projection of $\mathbf{w}^\top$ onto the space spanned by the data. We can see this using a singular value decomposition.

Proof. Let the singular value decomposition (SVD) of the data matrix be $\mathbf{X} = U \Sigma V^\top$, where Σ is the diagonal matrix of singular values σ_i. Substituting this into the term acting on $\mathbf{w}^\top$ yields

$$\begin{aligned}
\mathbf{X} \left(\gamma \mathbf{I}_D + \mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top &= U \Sigma V^\top \left(\gamma \mathbf{I}_D + V \Sigma^\top \Sigma V^\top\right)^{-1} V \Sigma^\top U^\top \\
&= U \Sigma V^\top \left[V (\gamma \mathbf{I}_D + \Sigma^\top \Sigma) V^\top\right]^{-1} V \Sigma^\top U^\top \\
&= U \underbrace{\Sigma (\gamma \mathbf{I}_D + \Sigma^\top \Sigma)^{-1} \Sigma^\top}_{\Lambda_\gamma} U^\top.
\end{aligned}$$

The inner matrix Λ_γ is diagonal with entries $(\Lambda_\gamma)_{ii} = \sigma_i^2 / (\gamma + \sigma_i^2)$. Taking the limit $\gamma \to 0$ we find

$$\lim_{\gamma \to 0} \frac{\sigma_i^2}{\gamma + \sigma_i^2} = \begin{cases} 1, & \text{if } \sigma_i \neq 0, \\ 0, & \text{if } \sigma_i = 0. \end{cases}$$

Thus, $\lim_{\gamma \to 0} \Lambda_\gamma$ acts as the identity on the subspace associated with non-zero singular values. Consequently, the full expression becomes $U_{\text{range}} U_{\text{range}}^\top$, which is the orthogonal projection operator onto the column space of $\mathbf{X}$. Therefore, in this limit, $\mathbf{w}_{\text{eff}}^\top$ is precisely the projection of $\mathbf{w}^\top$ onto the space spanned by the training data.

Expected Test Loss. In the heavily overparameterized limit, the distribution of $\mathbf{w} - \mathbf{w}_{\text{eff}}$ is straightforward. The j-th coefficient is zero if feature j appears at least once in the dataset, and is normally distributed with variance 1 if feature j does not appear. The contribution to the test loss is thus

$$\ell_{\text{test}} = \mathcal{E}_{\text{gen}} = \sum_j \Big(\mathbb{E}_{\text{test}}[x_j^2]\Big) \Big(\mathbb{E}_{\text{train}}\big[(\mathbf{w} - \mathbf{w}_{\text{eff}})_j^2\big]\Big).$$

Proof of Lemma 3.2 Here we derive the asymptotic expression for the expected number of coordinates that are active at least once across D i.i.d. samples under the sparse activation model.

Proof (learnable-coordinate count $K(D)$). In the sparse activation model, coordinate j is active in a single sample with probability $p_j = j^{-\alpha_1 - 1}$ and inactive with probability $1 - p_j$. The probability that coordinate j is never activated in D independent samples is

$$\mathbb{P}_{\text{never}}(j) = (1 - p_j)^D.$$

For $p_j \ll 1$, we use $(1 - p_j)^D \approx e^{-D p_j}$ as $D \to \infty$; this follows from $(1 - \frac{D p_j}{D})^D \to e^{-D p_j}$. Hence

$$\mathbb{P}_{\text{never}}(j) \approx \exp(-D p_j) = \exp\left(-D j^{-\alpha_1 - 1}\right).$$

Thus the probability that coordinate j is activated at least once is $1 - \exp(-D j^{-\alpha_1 - 1})$, and the expected number of such coordinates is

$$K(D) \;\approx\; \sum_{j=1}^{\infty} \left(1 - e^{-D j^{-\alpha_1 - 1}}\right).$$

Passing to the continuum limit, we obtain

$$K(D) \;\asymp\; \int_0^{\infty} \left(1 - e^{-D j^{-\alpha_1 - 1}}\right) dj.$$

We now perform the change of variables

$$t = Dj^{-(\alpha_1+1)} \quad \Longrightarrow \quad j = \left(\frac{D}{t}\right)^{\frac{1}{\alpha_1+1}}, \quad dj = -\frac{1}{\alpha_1+1} D^{\frac{1}{\alpha_1+1}} t^{-\frac{\alpha_1+2}{\alpha_1+1}}\, dt,$$

which yields

$$K(D) \asymp \frac{1}{\alpha_1+1} D^{\frac{1}{\alpha_1+1}} \int_0^\infty \left(1-e^{-t}\right) t^{-\frac{\alpha_1+2}{\alpha_1+1}}\, dt.$$

Let $\beta \equiv \frac{1}{\alpha_1+1} \in (0,1)$. By integration by parts,

$$\int_0^\infty (1-e^{-t})\, t^{-1-\beta}\, dt = \frac{1}{\beta} \int_0^\infty e^{-t} t^{-\beta}\, dt = \frac{\Gamma(1-\beta)}{\beta}.$$

Substituting $\beta = \frac{1}{\alpha_1+1}$ gives

$$K(D) \asymp \Gamma\left(1 - \frac{1}{\alpha_1+1}\right) D^{\frac{1}{\alpha_1+1}}.$$

This matches Eq. (5) in Lemma 3.2. □

Proof of Theorem 3.3 Here we give two derivations of Theorem 3.3: a discrete argument via the cutoff $K(D)$, which provides useful intuition, and a fully rigorous continuous argument.

Informal discrete argument via $K(D)$. By Lemma 3.2, when $\alpha_1 > 0$, at most $K(D)$ coordinates of the target vector $\mathbf{w}$ can be reliably estimated from D samples. All coordinates with indices $j > K(D)$ remain unobserved and therefore unmodeled. By the unmodeled-variance principle (Eq. (3)), the Bayes-optimal loss is

$$\ell_{\text{Bayes},D} \asymp \sum_{j>K(D)} j^{-\alpha_1-\alpha_2-2}.$$

Approximating the sum by an integral,

$$\begin{aligned}\sum_{j>K(D)} j^{-\alpha_1-\alpha_2-2} &\asymp \int_{K(D)}^\infty j^{-\alpha_1-\alpha_2-2}\, dj \\ &= \frac{1}{\alpha_1+\alpha_2+1} K(D)^{-(\alpha_1+\alpha_2+1)}.\end{aligned}$$

Substituting the scaling $K(D) \asymp D^{1/(\alpha_1+1)}$ from Eq. (5) yields

$$\ell_{\text{Bayes},D} \asymp D^{-(\alpha_1+\alpha_2+1)/(\alpha_1+1)},$$

which is Eq. (6). □

Rigorous proof (continuous argument). An equivalent expression for the Bayes-optimal loss can be written as a continuous expectation over coordinates, weighted by the probability that each coordinate is never activated. This leads to the integral representation

$$\ell_{\text{Bayes},D} = \int_0^\infty \exp\left(-Dj^{-\alpha_1-1}\right)\, j^{-(\alpha_1+\alpha_2+2)}\, dj, \tag{17}$$

We again apply the change of variables

$$t = Dj^{-(\alpha_1+1)} \quad \Longrightarrow \quad j = \left(\frac{D}{t}\right)^{\frac{1}{\alpha_1+1}}, \quad dj = -\frac{1}{\alpha_1+1} D^{\frac{1}{\alpha_1+1}} t^{-\frac{\alpha_1+2}{\alpha_1+1}}\, dt.$$

Substituting into (17) gives

$$\ell_{\text{Bayes},D} = \frac{1}{\alpha_1+1} D^{-\frac{\alpha_1+\alpha_2+1}{\alpha_1+1}} \int_0^\infty e^{-t}\, t^{\frac{\alpha_1+\alpha_2+1}{\alpha_1+1}-1}\, dt.$$

The remaining integral is exactly the Gamma function,

$$\int_0^\infty e^{-t} t^{\beta-1}\, dt = \Gamma(\beta),$$

with

$$\beta = \frac{\alpha_1 + \alpha_2 + 1}{\alpha_1 + 1}.$$

Hence

$$\ell_{\text{Bayes},D} = \frac{1}{\alpha_1 + 1} \Gamma\left(\frac{\alpha_1 + \alpha_2 + 1}{\alpha_1 + 1}\right) D^{-\frac{\alpha_1+\alpha_2+1}{\alpha_1+1}},$$

which matches the scaling law in Eq. (6) in the main text. When $\alpha_1 > 0$, sparsity reduces the effective number of observed coordinates to scale as $D^{1/(\alpha_1+1)}$, thereby becoming the dominant data bottleneck. This concludes the proof. □

Subtleties Concerning Theorem 3.3: Co-activation and Identifiability. A subtle complication arises when two rare coordinates co-activate, *i.e.*, are both nonzero in the same training example. In such cases, it becomes impossible to disentangle their individual contributions to the label, creating ambiguity in estimating the corresponding weights. The coordinates that activate exactly once have indices on the order of $j \sim D^{\frac{1}{\alpha_1+1}}$. Consequently, the total number of such coordinates also scales as $D^{\frac{1}{\alpha_1+1}}$. A nonzero number of pairwise co-activations is therefore likely whenever this quantity exceeds $D^{1/2}$ (by the birthday paradox), which occurs when $\alpha_1 < 1$. However, the number of co-activating coordinates itself scales only as

$$\min\left\{\frac{\left(D^{\frac{1}{\alpha_1+1}}\right)^2}{D},\ D^{\frac{1}{\alpha_1+1}}\right\} = \min\left\{D^{\frac{2}{\alpha_1+1}-1},\ D^{\frac{1}{\alpha_1+1}}\right\} = \min\left\{D^{\frac{1-\alpha_1}{\alpha_1+1}},\ D^{\frac{1}{\alpha_1+1}}\right\}.$$

Each such coordinate contributes variance on the order of $D^{\frac{-\alpha_1-\alpha_2-2}{\alpha_1+1}}$, so the total variance contributed by co-activating coordinates scales as

$$\min\left\{D^{\frac{-2\alpha_1-\alpha_2-1}{\alpha_1+1}},\ D^{\frac{-\alpha_1-\alpha_2-1}{\alpha_1+1}}\right\},$$

which vanishes as $D \to \infty$. Thus, while co-activation may create isolated ambiguity, it does not affect the asymptotic loss scaling in Theorem 3.3.

D Compute

This section provides an overview of the compute model we adopt for studying the compute-optimal frontier under training to completion:

$$C = ND \cdot \min\{N, D\}. \tag{18}$$

D.1 Computational Efficiency

Here, "compute" refers to the leading-order training cost, measured in FLOPs up to hardware-dependent constants. To approach the population-loss scaling studied in the main text, we focus on the setting in which the readout is trained to convergence. This can be done either with gradient-based methods, which are closer to practical training, or by directly computing the min-norm solution in Eq. (15).

D.1.1 Gradient-Based Methods

A single gradient step on the loss (14) costs $O(ND)$. The iteration complexity is controlled by the condition number of the feature Gram matrix, defined on its nonzero spectrum, $\kappa(\mathbf{\Phi}\mathbf{\Phi}^\top) = \lambda_{\max}(\mathbf{\Phi}\mathbf{\Phi}^\top)/\lambda^+_{\min}(\mathbf{\Phi}\mathbf{\Phi}^\top)$, where $\lambda^+_{\min}$ denotes the smallest nonzero eigenvalue. For plain GD, the number of steps scales linearly in κ, up to logarithmic accuracy factors. Accelerated first-order methods improve this dependence to $\sqrt{\kappa}$. We therefore write the leading gradient-based training cost as

$$C_{\text{grad}} \sim ND\,\kappa^p, \tag{19}$$

with $p = 1$ for GD and $p = 1/2$ for accelerated methods.

Condition number, κ. We estimate the condition number from the effective variance profile of the active features. The number of active coordinates in a dataset of size D is $K(D)$, so the readout effectively uses roughly $\min\{N, K(D)\}$ features. The variance explained by the smallest feature scales as $\min\{N, K(D)\}^{-(\alpha_1+\alpha_2+2)}$, while the largest is $O(1)$. Thus the effective condition number scales as

$$\kappa \sim \min\{N, K(D)\}^{\alpha_1+\alpha_2+2}. \tag{20}$$

Consequently, if the iteration complexity scales as κ^p, the number of optimization steps scales as $\min\{N, K(D)\}^{(\alpha_1+\alpha_2+2)p}$.

D.1.2 Direct Least-Squares Solve

Eq. (15) gives two equivalent expressions for $\boldsymbol{\theta}^\star$, related by a standard matrix identity. Using the first expression, $\mathbf{y}\,\boldsymbol{\Phi}^\top\mathbf{q}$, requires computing $\boldsymbol{\Phi}\boldsymbol{\Phi}^\top$ (cost N^2D), inverting it (cost N^3) to form $\mathbf{q}$, computing $\mathbf{y}\boldsymbol{\Phi}^\top$ (cost ND), and then multiplying the resulting row vector by $\mathbf{q}$ (cost N^2). The total cost is therefore $O(N^2D + N^3)$. By the same logic, using the second expression, $\mathbf{y}\,\mathbf{Q}\,\boldsymbol{\Phi}^\top$, costs $O(D^2N + D^3)$. The cheaper of these two options is

$$C_{\text{direct}} \sim ND \cdot \min\{N, D\}.$$

We summarize these estimates in Table 2, which compares the costs of gradient descent (GD), accelerated first-order methods, and direct solvers across the three regimes.

Table 2: Compute cost, normalized by ND, across the three parameter regimes. The GD and accelerated columns assume iteration complexity scaling as κ^p, with $p = 1$ for GD and $p = 1/2$ for accelerated first-order methods such as Nesterov acceleration. The direct-solve column corresponds to computing the min-norm least-squares solution in closed form.

Regime	GD	Accelerated	Direct solve
Heavily overparameterized, $N > D$	$D^{\frac{\alpha_1+\alpha_2+2}{\alpha_1+1}}$	$D^{\frac{\alpha_1+\alpha_2+2}{2(\alpha_1+1)}}$	D
Intermediate, $D > N > K(D)$	$D^{\frac{\alpha_1+\alpha_2+2}{\alpha_1+1}}$	$D^{\frac{\alpha_1+\alpha_2+2}{2(\alpha_1+1)}}$	N
Underparameterized, $K(D) > N$	$N^{\alpha_1+\alpha_2+2}$	$N^{\frac{\alpha_1+\alpha_2+2}{2}}$	N

D.2 Proxy Compute Model

Table 2 implies different computational scalings depending on both the solver and the regime.

Let $a = \alpha_1 + \alpha_2 + 2$ and $b = \alpha_1 + 1$, with $b > 0$. The comparison between direct solvers and first-order methods is solver-dependent. Since accelerated methods reduce the condition-number dependence from κ to $\kappa^{1/2}$, they are asymptotically cheaper than GD whenever $a > 0$, up to constants. The direct solve is cheaper than GD in the heavily overparameterized and intermediate regimes when $a/b > 1$, equivalently $\alpha_2 + 1 > 0$, while in the underparameterized regime the finite-variance condition $a > 1$ already guarantees that the direct solve is cheaper than GD.

The comparison with accelerated methods is stricter. In the heavily overparameterized regime, the direct solve beats acceleration only when $a/(2b) > 1$, equivalently $\alpha_2 > \alpha_1$. In the underparameterized regime, it beats acceleration only when $a/2 > 1$, equivalently $\alpha_1 + \alpha_2 > 0$. Thus direct solvers are uniformly cheapest under the stronger condition

$$\alpha_2 > \alpha_1 \qquad \text{and} \qquad \alpha_1 + \alpha_2 > 0. \tag{21}$$

There is also a mixed regime in which GD is the most expensive method, but the direct solve beats accelerated methods only in part of the phase diagram. For example, when $\alpha_1 > 0$ and

$$\max\{-1, -\alpha_1\} < \alpha_2 < \alpha_1, \tag{22}$$

acceleration is cheaper than the direct solve in the heavily overparameterized regime, while the direct solve is cheaper in the underparameterized regime. In the intermediate regime, the crossover occurs at $N_c(D) \sim D^{(\alpha_1+\alpha_2+2)/(2(\alpha_1+1))}$: direct solution is cheaper for $N < N_c(D)$, while acceleration is cheaper for $N > N_c(D)$.

For example, the representative choice $(\alpha_1, \alpha_2) = (1, 0.3)$ used in our experiments lies in this mixed regime: GD is asymptotically the most expensive method, while the cheapest method switches from acceleration in the heavily overparameterized regime to direct solution in the underparameterized regime, with a crossover between the two in the intermediate regime.

This solver dependence motivates using a simple unified compute proxy,

$$C = ND \cdot \min\{N, D\}, \tag{23}$$

rather than tying the main scaling analysis to a particular optimization algorithm. The qualitative conclusion we emphasize below is that sparsity shifts the compute-optimal allocation toward data, and this conclusion does not depend on these solver-dependent distinctions within reasonable compute models.

D.2.1 Robustness to Alternative Compute Models

The solver-dependent comparisons above change the mapping from (N, D) to compute, but not the basic allocation principle. The loss has the form

$$\ell(N, D) \sim N^{-\alpha_N} + D^{-\alpha_D}, \tag{24}$$

so the compute-optimal allocation balances the two terms:

$$N^{-\alpha_N} \sim D^{-\alpha_D} \qquad \Longrightarrow \qquad D \sim N^{\alpha_N/\alpha_D} = N^{\alpha_1+1}. \tag{25}$$

Thus sparsity fixes the relative allocation between data and model size independently of the particular solver. Different compute models only change how this balanced allocation scales with the total budget C.

For example, suppose that in the underparameterized branch the compute model takes the more general form

$$C \sim N^r D. \tag{26}$$

Combining this with $D \sim N^{\alpha_1+1}$ gives

$$N^*(C) \sim C^{1/(r+\alpha_1+1)}, \qquad D^*(C) \sim C^{(\alpha_1+1)/(r+\alpha_1+1)}, \tag{27}$$

and

$$\ell^*(C) \sim C^{-\alpha_N/(r+\alpha_1+1)}. \tag{28}$$

The direct-solve proxy used in Proposition 5.1 corresponds to $r = 2$, recovering Eq. (10). An accelerated first-order method in the same branch would instead have $r = 1 + \frac{\alpha_1+\alpha_2+2}{2}$, which changes the numerical compute exponent but leaves the balanced allocation $D^* \sim (N^*)^{\alpha_1+1}$ unchanged.

This distinction is especially useful in the mixed regime discussed above, where the cheapest solver can switch across the (N, D) plane. In that regime, acceleration is cheaper than the direct solve only in sufficiently model-heavy regions. The crossover in the intermediate regime occurs at $N_c(D) \sim D^{\frac{\alpha_1+\alpha_2+2}{2(\alpha_1+1)}}$. By contrast, the loss-balanced allocation satisfies $N_{\mathrm{bal}}(D) \sim D^{1/(\alpha_1+1)}$. In the mixed regime with $\alpha_1 + \alpha_2 > 0$, we have $\frac{1}{\alpha_1+1} < \frac{\alpha_1+\alpha_2+2}{2(\alpha_1+1)}$, so $N_{\mathrm{bal}}(D) \ll N_c(D)$ asymptotically. Therefore the compute-optimal allocation lies on the direct-solve side of the crossover, even though acceleration is cheaper in more heavily overparameterized regions.

For the representative experimental choice $(\alpha_1, \alpha_2) = (1, 0.3)$, this comparison gives $N_{\mathrm{bal}}(D) \sim D^{1/2}$ and $N_c(D) \sim D^{0.825}$. Thus the balanced allocation is well below the accelerated/direct crossover. Consequently, even under a piecewise cheapest-solver compute model, the asymptotic compute-optimal allocation is governed by the same direct-solve branch used in Proposition 5.1. More generally, alternative solver models can modify the compute exponent α_C, but they do not alter the qualitative conclusion that sparsity shifts the compute-optimal allocation toward data.

D.3 Compute-Optimal Allocation

Proposition 5.1 adopts the proxy compute budget in Eq. (23), $C = ND \cdot \min\{N, D\}$, and shows that the unique compute-optimal allocation lies in the underparameterized regime, $N < D$. The optimal

allocation and resulting loss scale as

$$\begin{aligned}
&\ell^*(C) \sim C^{-\alpha_C}, \\
&N^*(C) \sim C^{\alpha_D/(\alpha_N+2\alpha_D)} = C^{1/(\alpha_1+3)}, \qquad D^*(C) \sim C^{\alpha_N/(\alpha_N+2\alpha_D)} = C^{(\alpha_1+1)/(\alpha_1+3)}, \\
&\alpha_C := \frac{\alpha_N\alpha_D}{\alpha_N+2\alpha_D} = \frac{\alpha_1+\alpha_2+1}{\alpha_1+3} > 0.
\end{aligned} \tag{29}$$

D.3.1 Sketch of Proposition 5.1

Sketch (compute-optimal scaling). We optimize the loss under the compute model

$$C = ND \cdot \min\{N, D\},$$

Underparameterized Side ($N < D$). Here the compute constraint becomes

$$C = ND \cdot N = N^2 D.$$

Assuming the population loss when $\alpha_1 > 0$ takes the form

$$\ell(N, D) \sim N^{-\alpha_N} + D^{-\alpha_D},$$

we eliminate D using the compute constraint, $D = \frac{C}{N^2}$:

$$\ell(C, N) \;\sim\; N^{-\alpha_N} + \left(\frac{C}{N^2}\right)^{-\alpha_D} \;=\; N^{-\alpha_N} + C^{-\alpha_D} N^{2\alpha_D}.$$

For fixed C, the first term decreases with N while the second increases with N (since larger N forces smaller $D = C/N^2$). Thus the optimum balances the two contributions. Equating their scalings,

$$N^{-\alpha_N} \;\asymp\; C^{-\alpha_D} N^{2\alpha_D} \quad\Longleftrightarrow\quad N^{\alpha_N+2\alpha_D} \;\asymp\; C^{\alpha_D},$$

yields

$$N^*(C) \;\asymp\; C^{\frac{\alpha_D}{\alpha_N+2\alpha_D}}, \qquad D^*(C) = \frac{C}{(N^*(C))^2} \;\asymp\; C^{1-\frac{2\alpha_D}{\alpha_N+2\alpha_D}}.$$

At this optimum, both loss terms scale equally, so

$$\ell^*(C) \;\sim\; (N^*(C))^{-\alpha_N} \;\asymp\; C^{-\alpha_C}, \qquad \alpha_C = \frac{\alpha_N\alpha_D}{\alpha_N+2\alpha_D},$$

which gives the exponents reported in Eq. (29).

Overparameterized Side ($D < N$). In this case $\min\{N, D\} = D$ and the compute constraint becomes $C = ND^2$. Optimizing under this branch produces a candidate scaling that is not self-consistent with $D < N$, and is therefore ruled out by Remark 5.2, so the unique compute-optimal allocation lies in the underparameterized regime.

□

E Gradient Descent

Gradient descent (GD) provides a natural procedure for minimizing the training loss. In favorable settings, it converges to the trained-to-completion min-norm solution, though rare failures can occur due to sparsity as shown in the main text. Below, we provide a proof of Proposition 6.1 and a proof sketch for Theorem 6.2, which together summarize our main findings on GD, as reported in the main text.

E.1 Proof of Proposition 6.1

Proof (spectral bound for stability). Let $\mathbf{P}$ be the orthogonal projector onto $\mathrm{span}(\mathbf{X})$ and $k = \mathrm{rank}(\mathbf{X}\mathbf{X}^\top) \le K(D)$ its rank. Since $\mathbf{u}$ is drawn independently of $\mathbf{X}$, the restriction of $\mathbf{u}$ to this fixed k-dimensional subspace is an $N \times k$ matrix with i.i.d. $\mathcal{N}(0, 1/N)$ entries, so standard concentration gives $\mathbf{P}\mathbf{u}^\top\mathbf{u}\mathbf{P} = \mathbf{P} + o_{\mathbb{P}}(1)$ whenever $N \gg k$; the ambient dimension

M does not enter. This reduces (12) to a statement about $\mathbf{X}\mathbf{X}^\top$ alone in three steps. First, $\mathbf{u}^\top\mathbf{u}\,\mathbf{X}\mathbf{X}^\top$ has the same nonzero eigenvalues as the symmetric positive semidefinite matrix $(\mathbf{X}\mathbf{X}^\top)^{1/2}\,\mathbf{u}^\top\mathbf{u}\,(\mathbf{X}\mathbf{X}^\top)^{1/2}$, so $\lambda_{\max}$ is real and nonnegative. Second, $(\mathbf{X}\mathbf{X}^\top)^{1/2}$ has the same range as $\mathbf{X}\mathbf{X}^\top$, so $(\mathbf{X}\mathbf{X}^\top)^{1/2}\mathbf{P} = \mathbf{P}(\mathbf{X}\mathbf{X}^\top)^{1/2} = (\mathbf{X}\mathbf{X}^\top)^{1/2}$ and $\mathbf{P}$ may be inserted on either side of $\mathbf{u}^\top\mathbf{u}$ at no cost. Third, by near-isometry, $(\mathbf{X}\mathbf{X}^\top)^{1/2}\,\mathbf{P}\mathbf{u}^\top\mathbf{u}\mathbf{P}\,(\mathbf{X}\mathbf{X}^\top)^{1/2} = \mathbf{X}\mathbf{X}^\top + o_{\mathbb{P}}\big(\|\mathbf{X}\mathbf{X}^\top\|\big)$, and Weyl's inequality gives $\lambda_{\max}\big(\frac{1}{D}\mathbf{u}^\top\mathbf{u}\,\mathbf{X}\mathbf{X}^\top\big) = \lambda_{\max}\big(\frac{1}{D}\mathbf{X}\mathbf{X}^\top\big)\,(1 + o_{\mathbb{P}}(1))$. It therefore suffices to control $\lambda_{\max}\big(\frac{1}{D}\mathbf{X}\mathbf{X}^\top\big)$.

If empirical second moments concentrate, then $\frac{1}{D}\mathbf{X}\mathbf{X}^\top \approx \mathbf{\Sigma}_x$ with $\mathbf{\Sigma}_x = \mathbb{E}[\mathbf{x}\mathbf{x}^\top]$ diagonal and $\mathrm{Var}(x_j) = j^{-(\alpha_1+\alpha_2+2)}$, whose top eigenvalue equals $\mathrm{Var}(x_1) = 1$. Although $\mathbf{\Sigma}_x$ is diagonal, $\frac{1}{D}\mathbf{X}\mathbf{X}^\top$ need not be. The off-diagonal entries have mean zero and concentrate entrywise; we assume that their aggregate contribution is subleading, so they do not change the leading $O(1)$ scale of the top eigenvalue governed by the diagonal part. The only potential obstruction is an anomalously large diagonal entry caused by a single rare activation at a very large index when $\alpha_2 < -1$ (so activated amplitudes grow with j); in that case $\alpha_1 > -\alpha_2 - 1 > 0$.

Let $j_{\max}$ be the largest index that activates at least once in the dataset. For a given threshold J, the expected number of activations at indices larger than J is

$$D\sum_{j>J}\mathbb{P}(x_j \neq 0) = D\sum_{j>J} j^{-(\alpha_1+1)} \asymp \frac{1}{\alpha_1} D\,J^{-\alpha_1}.$$

Equivalently, writing

$$N_{>J} := \sum_{d=1}^{D}\sum_{j>J}\mathbf{1}\{x_j^{(d)} \neq 0\},$$

a Poisson approximation yields

$$\mathbb{P}(N_{>J} = 0) \approx \exp\!\left(-\frac{D}{\alpha_1}J^{-\alpha_1}\right), \qquad \mathbb{P}(j_{\max} \le J) = \mathbb{P}(N_{>J} = 0) \approx \exp\!\left(-\frac{D}{\alpha_1}J^{-\alpha_1}\right),$$

and hence

$$\mathbb{P}(j_{\max} = J) \approx D\,J^{-(\alpha_1+1)}\exp\!\left(-\frac{D}{\alpha_1}J^{-\alpha_1}\right).$$

In particular, $j_{\max} \asymp D^{1/\alpha_1}$ with high probability.

Conditional on an activation at $j_{\max}$, the corresponding diagonal contribution to $\frac{1}{D}\mathbf{X}\mathbf{X}^\top$ is of order

$$\frac{1}{D}x_{j_{\max}}^2 = \frac{1}{D}j_{\max}^{-(\alpha_2+1)} \sim D^{-(\alpha_1+\alpha_2+1)/\alpha_1}.$$

But, given that $\alpha_2 < -1$, and since $\alpha_1 + \alpha_2 + 1 > 0$, we have $D^{-(\alpha_1+\alpha_2+1)/\alpha_1} \to 0$, so such rare spikes cannot dominate the $O(1)$ contribution from the leading coordinates (in particular $j = 1$). Therefore $\lambda_{\max}\big(\frac{1}{D}\mathbf{X}\mathbf{X}^\top\big) = 1 + o_{\mathbb{P}}(1)$, and the same holds for $\lambda_{\max}\big(\frac{1}{D}\mathbf{u}^\top\mathbf{u}\,\mathbf{X}\mathbf{X}^\top\big)$ under near-isometry. The conclusion of Proposition 6.1 then follows from the usual spectral stability condition on the step size. □

E.2 Sketch of Theorem 6.2

Sketch (rare-activation spike). We focus on the event that a single "late" coordinate activation produces a spiked top eigenvalue. Write the data matrix as $\mathbf{X} = [\,\mathbf{X}_{\mathrm{reg}}\ \mathbf{x}_{\mathrm{spec}}\,]$, where $\mathbf{X}_{\mathrm{reg}} \in \mathbb{R}^{M\times(D-1)}$ contains the regular samples and $\mathbf{x}_{\mathrm{spec}} \in \mathbb{R}^M$ is a special sample containing a single unusually large activation. Then

$$\mathbf{X}\mathbf{X}^\top = \mathbf{X}_{\mathrm{reg}}\mathbf{X}_{\mathrm{reg}}^\top + \mathbf{x}_{\mathrm{spec}}\mathbf{x}_{\mathrm{spec}}^\top.$$

In a spiked covariance scenario, the top eigenvector aligns with $\mathbf{x}_{\mathrm{spec}}$ and the corresponding eigenvalue is bounded below by the Rayleigh quotient

$$\lambda_{\max}(\mathbf{X}\mathbf{X}^\top) \ge \frac{\mathbf{x}_{\mathrm{spec}}^\top\mathbf{X}\mathbf{X}^\top\mathbf{x}_{\mathrm{spec}}}{\|\mathbf{x}_{\mathrm{spec}}\|_2^2} = \frac{\mathbf{x}_{\mathrm{spec}}^\top\mathbf{X}_{\mathrm{reg}}\mathbf{X}_{\mathrm{reg}}^\top\mathbf{x}_{\mathrm{spec}}}{\|\mathbf{x}_{\mathrm{spec}}\|_2^2} + \|\mathbf{x}_{\mathrm{spec}}\|_2^2. \tag{30}$$

Under the finite-variance condition $\alpha_1 + \alpha_2 + 1 > 0$, the first term in (30) is $o(D)$ at the rare-spike scale, so the condition for GD instability, $\lambda_{\max}\big(\mathbf{X}\mathbf{X}^\top\big) > \frac{2D}{\eta}$, is to leading order in D equivalent to requiring

$$\|\mathbf{x}_{\text{spec}}\|_2^2 \;\gtrsim\; \frac{2D}{\eta}. \tag{31}$$

In our model, a single-coordinate activation at index j has magnitude $\|\mathbf{x}_{\text{spec}}\|_2^2 \approx x_j^2 = j^{-(\alpha_2+1)}$. Hence (31) requires

$$j \;\gtrsim\; j^* := \left(\frac{2D}{\eta}\right)^{\frac{1}{-\alpha_2-1}} \qquad (\alpha_2 < -1). \tag{32}$$

Finally, let $N_{>j^*}$ denote the (random) number of activations with index larger than j^* in the dataset. By definition of j^*, the rare-spike divergence event is the existence of at least one activation with index $> j^*$, so $\mathbb{P}_{\text{rare}}(\text{diverge}) \asymp \mathbb{P}(N_{>j^*} \geq 1)$.

The expected number of activations at indices larger than j^* is

$$\mathbb{E}[N_{>j^*}] = D \sum_{j>j^*} \mathbb{P}(x_j \neq 0) = D \sum_{j>j^*} j^{-(\alpha_1+1)} \;\asymp\; \frac{1}{\alpha_1} D\,(j^*)^{-\alpha_1}.$$

Equivalently (as in Subsection E.1), a Poisson approximation gives $\mathbb{P}(N_{>j^*} \geq 1) = 1 - \exp(-\mathbb{E}[N_{>j^*}]) \asymp \mathbb{E}[N_{>j^*}]$ when $\mathbb{E}[N_{>j^*}]$ is small, hence $\mathbb{P}(N_{>j^*} \geq 1) \asymp \frac{1}{\alpha_1} D\,(j^*)^{-\alpha_1}$.

Substituting (32) gives

$$\mathbb{P}_{\text{rare}}(\text{diverge}) \asymp \frac{1}{\alpha_1} D \left(\frac{2D}{\eta}\right)^{-\frac{\alpha_1}{-\alpha_2-1}} \asymp D^{-\frac{\alpha_1+\alpha_2+1}{-\alpha_2-1}},$$

which is Eq. (13) up to η-dependent constants. □

E.2.1 Empirical Considerations for Theorem 6.2

We discuss the practical difficulty of empirically validating Theorem 6.2's rare-spike prediction $\mathbb{P}_{\text{rare}}(\text{diverge}) \asymp D^{-\nu}$, and report our finite-$D$ measurements, which are consistent with the theorem's qualitative form.

A clean validation would estimate $\mathbb{P}\big(\lambda_{\max}(\mathbf{X}\mathbf{X}^\top/D) > \epsilon\big)$ for fixed threshold $\epsilon = 2/\eta$ across a range of D values spanning the asymptotic regime, then fit the slope on log-log axes and compare to the predicted exponent $\nu = (\alpha_1 + \alpha_2 + 1)/(-\alpha_2 - 1)$. However, this represents a statistical challenge: the theorem characterizes a tail probability, and at any given D, accurate estimation of $\mathbb{P}_{\text{rare}}(\text{diverge}) = p$ requires $\Theta(1/p)$ independent samples to observe events at the relevant rate, with sample size scaling further with ν for accurate exponent estimation. For the theorem's asymptotic regime to dominate observation, D must be large enough that finite-D corrections to the spectral concentration are negligible compared to the leading $D^{-\nu}$ behavior; in our experiments, this asymptotic crossover occurs beyond $D \approx 10^4$. Combining these requirements, a quantitative validation at a predicted exponent $\nu = 9$ (for chosen parameters $\alpha_1 = 2$, $\alpha_2 = -1.2$) would require roughly 10^9 to 10^{12} independent dataset samples per D value—many orders of magnitude beyond computational tractability with eigenvalue computation on $D \times D$ Gram matrices.

We perform experiments in which we sweep $D \in \{500, 1000, 2000, 5000, 10000\}$ at fixed threshold $\epsilon = 1.0005$ and feature dimension $M = 5000$, sampling between 500 and 4000 independent dataset replicates per D. We chose $\alpha_1 = 2,\ \alpha_2 = -1.2$ (giving $\nu = 9$) as a representative point well inside the regime $\alpha_2 < -1$ and $\alpha_1 + \alpha_2 + 1 > 0$ where Theorem 6.2 applies (smaller values of ν are more easily detected in finite samples but tend to give less clean separation between tail-event probabilities and bulk concentration). The threshold ϵ was chosen empirically so that the smallest-D measurement gives a moderate (non-saturated) failure rate. Our results are shown in Table 3.

The decay spans more than three orders of magnitude in $\mathbb{P}(\text{diverge})$ over an order of magnitude in D. Computing local exponents from successive D pairs gives $\nu_{\text{local}} \approx 1.5,\ 2.7,\ 4.5$, increasing monotonically—consistent with finite-D corrections that vanish as the asymptotic regime is approached.

Table 3: Empirical divergence probability as a function of dataset size D, with $\alpha_1 = 2$, $\alpha_2 = -1.2$, $M = 5000$, threshold $\epsilon = 1.0005$.

D	failures / seeds	$\mathbb{P}(\text{diverge})$
500	185/500	0.370
1000	132/1000	0.132
2000	40/2000	0.020
5000	1/3000	0.00033
10000	0/4000	< 0.00025

Two observations matter for the comparison to theory. First, the empirical $\mathbb{P}(\text{diverge})$ exhibits a clean, monotone power-law decay with D—consistent in functional form and direction with Theorem 6.2. Second, the empirical exponent at the largest measurable D ($\nu_{\text{local}} \approx 4.5$) is approaching, but has not reached, the asymptotic prediction $\nu = 9$. Both observations are consistent with the theorem: the asymptotic exponent is a statement about $D \to \infty$, and the local empirical exponent's monotonic increase suggests we are inside the finite-D regime where the asymptotic rate has not yet emerged. Distinguishing the asymptotic exponent $\nu = 9$ from any other large-ν value (e.g., $\nu = 6$ or $\nu = 12$) requires sample sizes and D ranges substantially beyond what is computationally accessible with this experimental design. We conclude that our measurements are quantitatively non-contradictory with the theorem and qualitatively confirm its prediction of a sharp power-law decay in the divergence probability with dataset size.

F Experiments on the Sparse Activation Model

F.1 Min-norm least-squares solutions via gradient-based optimization

Our scaling-law theory predicts the test loss of the min-norm random-feature readout as N and D vary. We use two independent methods to recover this solution:

1. **Closed-form pseudoinverse.** For the linear regression problem $\min_{\boldsymbol{\theta}} \frac{1}{2D}\|\boldsymbol{\Phi}^\top \boldsymbol{\theta} - \mathbf{y}^\top\|_2^2$ on (linear or ReLU) random features, the min-norm least-squares solution is $\boldsymbol{\theta}^\star = (\boldsymbol{\Phi}^\top)^+ \mathbf{y}^\top$ (equivalently $\boldsymbol{\theta}^{\star\top} = \mathbf{y}\boldsymbol{\Phi}^+$), computed via SVD-based pseudoinverse with a relative tolerance of 10^{-10}–10^{-9}.
2. **Nesterov-accelerated GD with adaptive restart** [73, 74]. The readout θ is initialized at zero and trained on the empirical mean-squared error with Nesterov momentum and gradient-based adaptive restart. From zero initialization, the converged solution is provably the same min-norm least-squares solution as the closed-form one.

Figures 2 and 3 are computed using the closed-form pseudoinverse, while Figure 4 is computed using the Nesterov-accelerated solver.

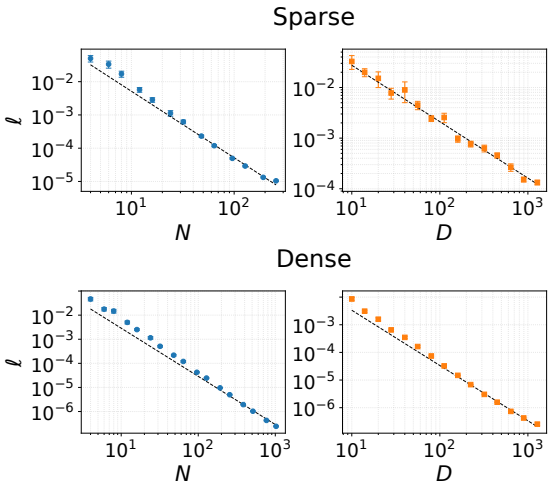

Figure 6: **Two-exponent scaling validates linear theory.** Test loss scaling under the linear feature map $\phi(\mathbf{x}) = \mathbf{u}\mathbf{x}$, computed via the closed-form min-norm least-squares solution; $(\alpha_1, \alpha_2) = (1.0, 0.3)$, 20 seeds. Top: sparse; bottom: dense. Left: N-sweep at $D = 50{,}000$; right: D-sweep at $N = 16{,}000$. A single exponent $\alpha \approx 2.0$ (dashed lines, jointly fitted with separate intercepts) describes the sparse N-, dense N-, and dense D-sweeps, in close agreement with the linear-theory prediction $\alpha_N = \alpha_1 + \alpha_2 + 1 = 2.30$. The sparse D-sweep requires a shallower fit, $\alpha_D \approx 1.11$, matching the predicted sparse exponent $\alpha_D = (\alpha_1 + \alpha_2 + 1)/(\alpha_1 + 1) = 1.15$ and breaking the dense-baseline symmetry $\alpha_N = \alpha_D$.

The closed-form and Nesterov approaches give the same qualitative two-regime structure, supporting the interpretation that the observed exponent asymmetry is primarily a property of the regression problem rather than of a particular solver. In the linear case (Figure 6), the joint exponent across the sparse N-sweep,

dense N-sweep, and dense D-sweep recovers $\alpha \approx 2.0$, in close agreement with the linear-theory prediction $\alpha_N = \alpha_1 + \alpha_2 + 1 = 2.30$, while the sparse D-sweep gives $\alpha_D \approx 1.11$, matching the theoretical prediction $\alpha_D = (\alpha_1 + \alpha_2 + 1)/(\alpha_1 + 1) = 1.15$. The asymmetry ratio $\alpha_N/\alpha_D \approx 1.80$ is close to the predicted $2.30/1.15 = 2.00$, with a residual gap plausibly attributable to finite-size corrections and possible crossover effects near the double-descent peak. Under ReLU (Figures 7 and 4) both methods give the same qualitative structure: a joint exponent describing three of the four sweeps and a shallower sparse D-exponent breaking the dense-baseline symmetry, with the magnitudes $\alpha_N/\alpha_D \approx 1.13$ (closed-form) and ≈ 1.23 (Nesterov). These finite-size corrections are roughly consistent across the linear and ReLU cases, indicating that nonlinearity reduces the magnitude of the asymmetry but does not alter its qualitative structure.

Setup. All experiments use the data-generating model defined in Section 2.1, with sparse parameters $(\alpha_1, \alpha_2) = (1.0, 0.3)$ and ambient dimension $M = 10{,}000$–$20{,}000$. The teacher is $y(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$ with $w_j \sim \mathcal{N}(0, 1)$ i.i.d. The first-layer weights $\mathbf{u} \in \mathbb{R}^{N \times M}$ are drawn i.i.d. from $\mathcal{N}(0, 1/N)$ and held frozen throughout training. The dense baseline uses Gaussian inputs with per-coordinate variance $j^{-(\alpha+1)}$, $\alpha = \alpha_1 + \alpha_2 + 1$, matching the underparameterized exponent of the sparse model so that any asymmetry in scaling is attributable purely to sparsity. Test loss is evaluated on n_{test} fresh samples drawn from the same distribution.

For each (N, D) configuration we run multiple seeds (varying both data sampling and the random first layer) and report mean test loss with ± 1 standard error. Power-law exponents are fitted on log-log axes by least squares, restricted to the asymptotic tail (last 4–6 values of N or D) to mitigate finite-size deviations. As a check, we also compare full-batch GD with minibatch SGD at small learning rate; the resulting trajectories and scaling exponents agree closely, suggesting that the sparse scaling laws reported here are not driven primarily by minibatch gradient noise, but by the underlying optimization/statistical structure.

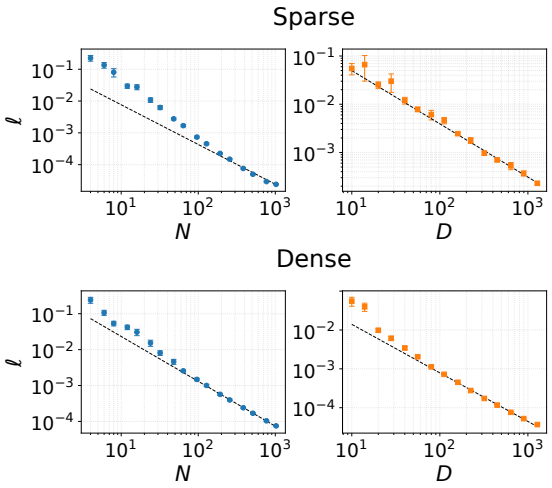

Figure 7: **Asymmetry persists under nonlinearity (closed-form).** Test loss scaling under the ReLU feature map $\phi(\mathbf{x}) = \sigma(\mathbf{ux})$, computed via the closed-form min-norm least-squares solution; $(\alpha_1, \alpha_2) = (1.0, 0.3)$, 20 seeds. Top: sparse; bottom: dense. Left: N-sweep at $D = 50{,}000$; right: D-sweep at $N = 16{,}000$. A single exponent $\alpha \approx 1.25$ (dashed lines, jointly fitted with separate intercepts) describes the sparse N-, dense N-, and dense D-sweeps. The sparse D-sweep requires a shallower fit, $\alpha_D \approx 1.10$, breaking the dense-baseline symmetry $\alpha_N = \alpha_D$ predicted by linear theory; the smaller magnitude of the asymmetry compared to the linear case (Figure 6) reflects spectral smoothing under nonlinear feature maps.

Experiment 1: Linear random features. The forward map is $\hat{y} = \theta^\top(\mathbf{ux})$. The min-norm readout is solved in float64 via `torch.linalg.pinv` with `rtol` $= 10^{-10}$ to handle rank-deficient feature matrices robustly. We sweep $N \in \{4, 6, 8, 12, 16, 24, 32, 48, 64, 96, 128, 192, 256\}$ at fixed $D = 50{,}000$, and $D \in \{10, 14, 20, 28, 40, 56, 80, 112, 160, 224, 320, 448, 640, 896, 1280\}$ at fixed $N = 16{,}000$, with 20 seeds per configuration and $n_{\text{test}} = 50{,}000$. Fits use the last 6 points of each sweep.

Experiment 2: ReLU random features. The forward map is $\hat{y} = \theta^\top \sigma(\mathbf{ux})$ with σ the ReLU activation. Two solver variants are used:

Closed-form (pinv). Identical to Experiment 1, with ReLU applied to the features before the pseudoinverse. Same sweep grids, 20 seeds, and $n_{\text{test}} = 50{,}000$.

Iterative (Nesterov). The ReLU features $\mathbf{\Phi} = \sigma(\mathbf{uX}) \in \mathbb{R}^{N \times D}$ are precomputed once in float64 on GPU. The step size is set to $\eta = 1/\lambda_{\max}(\mathbf{\Phi}\mathbf{\Phi}^\top/D)$, where $\lambda_{\max}$ is estimated by 50 steps of power iteration. Nesterov updates with O'Donoghue–Candès adaptive restart [74] are applied until the

relative gradient norm satisfies $\|\nabla\| < 10^{-9}\|\nabla_0\|$, with a maximum of 5×10^5 iterations. We sweep $N \in \{4, 8, 16, 32, 64, 128, 256\}$ at fixed $D = 50{,}000$, and $D \in \{10, 20, 40, 80, 160, 320, 640, 1280\}$ at fixed $N = 8000$, with 5 seeds per configuration and $n_{\text{test}} = 20{,}000$. Fits use the last 4 points of each sweep.

Hardware and reproducibility. Experiments 1 and 2 were run on a single NVIDIA T4 or A100 GPU (Google Colab). Float64 precision is used throughout the optimization and pseudoinverse computations; data sampling and feature evaluation use float32 with TF32 matrix multiplications enabled. The first layer $\mathbf{u}$ is regenerated per seed.

G Experiments on Pile Token Statistics

Data. Each sample is a single token, with coordinates indexed by decreasing frequency, so the input is $x_j = j^{-(\alpha_2+1)/2}$ on the active coordinate and zero elsewhere, with $\alpha_2 = 0.3$; coordinate j is active with its measured frequency p_j, so $\mathbb{E}[x_j^2] = p_j\, j^{-(\alpha_2+1)}$. The target is $y = \mathbf{w}^\top \mathbf{x}$ with $w_j \sim \mathcal{N}(0, 1)$ i.i.d. The dense baseline draws each x_j from a zero-mean Gaussian with the same $\mathbb{E}[x_j^2]$, with every coordinate active in every sample. $K(D)$ is obtained by counting distinct coordinates among D samples. The double-descent experiment uses the Pythia tokenizer on a 10,000-document sample of the Pile, truncated to the 4000 most frequent types; the exponent fits use the `o200k_base` tokenizer over 1.48×10^8 tokens, which contain 147,775 distinct types.

Double-descent peak. We take random features $\mathbf{u} \in \mathbb{R}^{N \times M}$ with entries drawn i.i.d. from $\mathcal{N}(0, 1/N)$ and compute the min-norm least-squares solution, with 4 seeds, for 18 values of D log-spaced from $N/4$ to 8000. At $N = 600$ the sparse peak is at $D = 1231$ against $K(D) = N$ at $D = 1208$, and the dense peak at 610; at $N = 1200$ the sparse peak is at 3046 against 3184, and the dense peak at 1160 (Figure 5). All four lie within 5% of their predicted values.

Linear random features. With one active token per sample, the design is fixed by the per-coordinate counts, and the population loss $\sum_j \mathbb{E}[x_j^2]\,(w_j - \hat{w}_j)^2$ is computed exactly. All four sweeps span the same effective rank, 90 to 4300 (10 points, 1.68 decades): the N-sweeps take N over this range at $D = 3.28 \times 10^7$ (sparse, $K = 129{,}017$) and $D = 51{,}600$ (dense); the D-sweeps run at $N = 45{,}000$ with D chosen so that $K(D)$ (sparse, $D = 103$ to 10,073) or D (dense) covers the range. We use 5 seeds for each N-sweep, and 30 for the sparse and 15 for the dense D-sweep. We fit each exponent as the log-log slope of the geometric mean over seeds; brackets are 95% bootstrap intervals over seeds.

	α_N	α_D	$\alpha_N - \alpha_D$
sparse	$1.321\,[1.301, 1.341]$	$1.259\,[1.227, 1.294]$	$+0.062\,[+0.021, +0.101]$
dense	$1.242\,[1.206, 1.281]$	$1.262\,[1.247, 1.279]$	$-0.020\,[-0.060, +0.020]$

The difference of the two gaps is $+0.082\,[+0.026, +0.133]$, positive in 99.8% of bootstrap samples. Refitting on sub-windows (dropping the lowest or highest point, or keeping the lower or upper half) gives a positive difference in all four cases (0.048 to 0.093), with the interval excluding zero in two. Along the sparse D-sweep, the ratio of the measured loss to the variance of the never-observed coordinates lies between 0.86 and 1.11, with nothing fitted. Figure 8 plots $(\ell(z)/\ell(z_0))\,(z/z_0)^{\alpha_N} - 1$ along both sweeps, with $z = N$ or D, z_0 the first point of the sweep, and α_N the model-size exponent fitted for each design. For a power law $\ell \propto z^{-\alpha}$ this quantity equals $(z/z_0)^{\alpha_N - \alpha} - 1$, so the N-sweep is flat by construction and the D-sweep rises if $\alpha_D < \alpha_N$ and falls if $\alpha_D > \alpha_N$. The sparse D-sweep rises and the dense one falls slightly.

Neural network in the lazy regime. We also study scaling on Pile token statistics in a network trained in the lazy regime. The network is $\mathbf{x} \to \mathbf{u}\mathbf{x} \in \mathbb{R}^N \to \mathrm{ReLU}(H) \to \mathrm{ReLU}(H) \to$ scalar, in the neural tangent kernel (NTK) parameterization [75], with $n_{\text{par}} = NH + H^2 + H$ trainable parameters. It is linearized at initialization, so training to completion is min-norm regression in the empirical tangent kernel, solved in closed form. Without a random projection, distinct one-hot inputs are orthogonal, and because the network has no biases its tangent kernel is positively homogeneous, $\Theta(c\mathbf{x}, c'\mathbf{x}') = cc'\,\Theta(\mathbf{x}, \mathbf{x}')$ for $c, c' > 0$. On inputs $\mathbf{x} = a_j \mathbf{e}_j$, with $a_j = j^{-(\alpha_2+1)/2}$, it therefore takes the form $\Theta = \kappa_0\, \mathbf{a}\mathbf{a}^\top + (\kappa_1 - \kappa_0)\,\mathrm{diag}(a_j^2)$, a rank-one term plus a diagonal, with depth and width entering only through the two constants κ_0 and κ_1 (exactly at infinite width, up to fluctuations

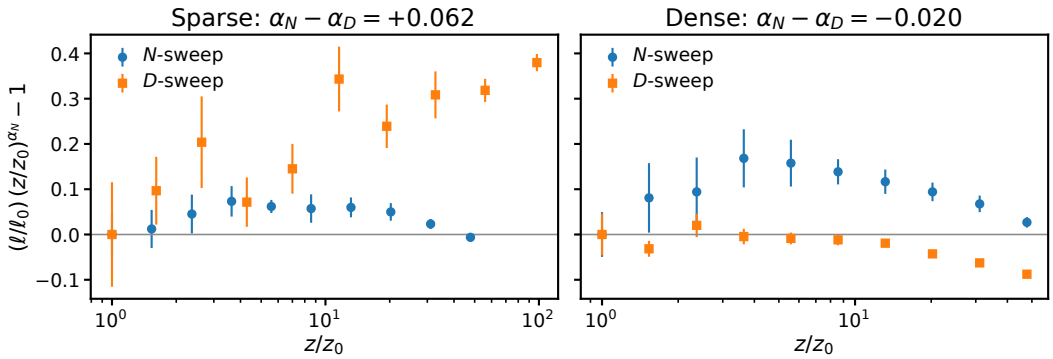

Figure 8: **Asymmetry grows with sparsity on Pile token statistics (linear random features).** Each sweep is plotted as $(\ell(z)/\ell(z_0))\,(z/z_0)^{\alpha_N} - 1$, with $z = N$ or D and α_N the fitted model-size exponent of that design; for a power law $\ell \propto z^{-\alpha}$ this equals $(z/z_0)^{\alpha_N - \alpha} - 1$. The N-sweep then has zero fitted slope, and the log-slope of the D-sweep is $\alpha_N - \alpha_D$. Left: sparse, $\alpha_N - \alpha_D = +0.062$; right: dense, -0.020. All sweeps span the same effective rank, 90 to 4300. Error bars are ± 1 standard error over seeds (5 for the N-sweeps, 30 sparse and 15 dense for the D-sweeps).

at finite width). The min-norm solution then interpolates the K observed tokens exactly and predicts every unobserved token only along the single direction $\mathbf{a}$, assigning each the same multiple of its amplitude a_j, so the number of target components the model can fit is K, set by the data; width and depth change only κ_0 and κ_1, and the loss has no dependence on model size. With the projection, token j enters as $\mathbf{u}_j \in \mathbb{R}^N$, distinct tokens overlap at $O(N^{-1/2})$, and the first layer resolves at most N directions, so N acts as a capacity parameter. On the data axis ($N = 4000$, $D = 100$ to 10,000, 3 seeds), the loss tracks the variance of the unobserved coordinates, with $\alpha_D = 1.16\,[0.70, 1.40]$, $1.28\,[1.12, 1.47]$, $1.29\,[1.17, 1.41]$, $1.27\,[1.15, 1.43]$ at $H = 8$, 32, 128, 1024 (Figure 9, left). On the model-size axis, only $H = 8$ keeps n_{par} below K (n_{par}/K at the largest N is 1.01, 4.09, 17.38, 215.49 for the four widths). For the sparse case we fix the corpus so that $K \approx 12{,}000$ distinct tokens are observed, and we run the dense baseline at $D = 12{,}000$ samples, so that n_{par}/D in the dense baseline equals n_{par}/K in the sparse case at every N. Fitting $N = 60$ to 513 (0.93 decades, the values with $n_{\mathrm{par}} \le K/2$; 3 sparse and 2 dense seeds), the sparse design gives $\alpha_N = 1.05\,[0.88, 1.16]$ and the dense baseline $-0.07\,[-0.32, +0.17]$ (Figure 9, right). Both exponents are fitted after dividing the loss by $1/(1 - n_{\mathrm{par}}/D)$, the factor by which least squares inflates the risk as n_{par} approaches the number of constraints; this factor grows from 1.05 to 1.53 over the fitted range and is the same for both designs, so it shifts the two exponents equally (0.88 and -0.24 without it) and leaves their difference, $+1.13\,[+0.79, +1.47]$, unchanged. We could not obtain a usable dense data exponent in

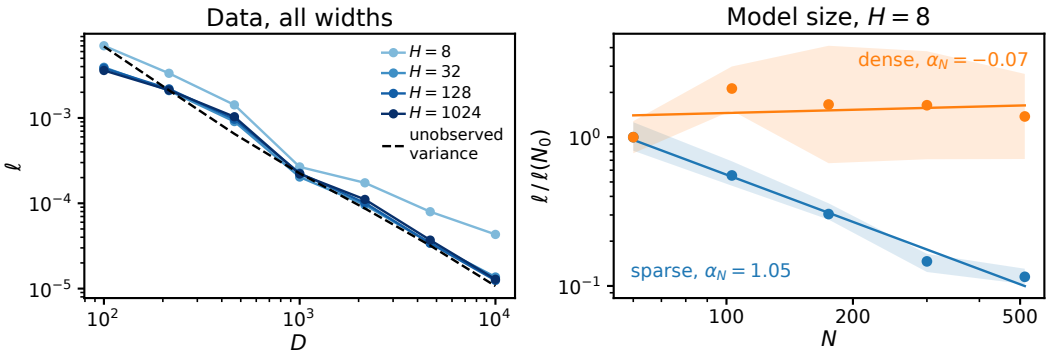

Figure 9: **A network trained on Pile token statistics in the lazy regime.** A depth-3 ReLU network of width H on an N-dimensional random projection $\mathbf{ux}$, linearized at initialization. Left: loss versus D at $N = 4000$ for four widths (3 seeds), against the variance of the coordinates not observed among the D samples (dashed, not fitted). Right: loss versus N at $H = 8$, with $K \approx 12{,}000$ observed coordinates for the sparse design and $D = 12{,}000$ for the dense baseline, corrected by $1/(1 - n_{\mathrm{par}}/D)$ and normalized to the first point; lines are power-law fits and shading spans seeds (3 sparse, 2 dense).

this setting, so we do not compare the asymmetry between the two designs. The results nonetheless show that the loss along D tracks the variance of the unobserved tokens at every width, and indicate that in this architecture a decrease of the loss with model size may require sparsity, which deserves further study. As checks, at depth one the tangent model reproduces the linear random-feature model to a relative difference of 1.3×10^{-13}, and the seed-to-seed spread of the empirical kernel falls from 0.38 to 0.08 between $H = 32$ and 2048, as it approaches its infinite-width limit.

Hardware and reproducibility. The double-descent experiment runs on CPU in Google Colab, and the linear random-feature and lazy-network sweeps each run on a single GPU in under an hour. Gram matrices, kernels and solves are computed in float64, and the random projection is drawn in float32 on the GPU, so the exponents reproduce exactly on fixed hardware and shift at most in the second decimal across devices.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims we make are supported by both theory and experiment, and are empirically shown to generalize beyond our settings and assumptions.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We provide a detailed discussion of the limitations of our work, including potential violations of our assumptions. These limitations also suggest several interesting directions that deserve dedicated follow-up work in their own right.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We provide the full set of assumptions and complete and correct proofs for every theoretical result we present.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide detailed descriptions of our methodology as well as code for the relevant experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We provide codes for our main experiments.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide experimental details for all experiments in the appendices.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report statistical errors when relevant.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes].

   Justification: Appendices F and G specify the hardware we used: a single NVIDIA T4 or A100 GPU on Google Colab and a CPU for the double-descent peak experiment.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We adhere to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

    Justification: Our paper is primarily theoretical, and we expect it to have limited negative societal impact. A possible misuse is that our conclusion that compute favors more data in sparse regimes could be used to guide the training of harmful models on sparse but dangerous information. We view this risk as unlikely, since the work analyzes theory rather than providing a practical recipe for building such systems.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We use the Pile [16] through two Hugging Face releases, `NeelNanda/pile-10k` and `monology/pile-uncopyrighted`, together with the Pythia-70M-deduped tokenizer and the `o200k_base` tokenizer of `tiktoken`, all used under their licenses.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We release the code for every experiment and figure at `https://github.com/jsous/asym-scaling`.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A].

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.