The Interplay of Data Structure and Imbalance in the Learning Dynamics of Diffusion Models

Flavio Nicoletti
DSAI
Chalmers and University of Gothenburg
flavion@chalmers.se

Chenxiao Ma
DSAI
Chalmers and University of Gothenburg
mache@chalmers.se

Enrico Ventura
International School of Advanced Studies (SISSA)
eventura@sissa.it

Luca Saglietti
Department of Computing Sciences
Bocconi University
luca.saglietti@unibocconi.it

Stefano Sarao Mannelli
DSAI
Chalmers and University of Gothenburg
s.saraomannelli@chalmers.se

Abstract

Real-world datasets are inherently heterogeneous, yet how per-class structural differences and sampling imbalance shape the training dynamics of diffusion models—and potentially exacerbate disparities—remains poorly understood. While models typically transition from an initial phase of generalization to memorizing the training set, existing theory assumes homogeneous data, leaving open how class imbalance and heterogeneity reshape these dynamics. In this work, we develop a high-dimensional analytical framework to study class-dependent learning in score-based diffusion models. Analyzing a random-features model trained on Gaussian mixtures, we derive the feature-covariance spectrum to characterize per-class generalization and memorization times. We reveal the explicit hierarchy governing these dynamics: class variance is the primary determinant of learning order—consistently favoring higher-variance classes—while centroid geometry and sampling imbalance play secondary roles. Specifically, sampling imbalance acts as a modulator that can reverse this ordering and, under strong imbalance, forces minority classes to acquire distinct, delayed inference times during backward diffusion. Together, these results suggest that diffusion models can memorize some classes while others remain insufficiently learned. We validate our theoretical predictions empirically using U-Net models trained on Fashion MNIST.

1 Introduction

In generative models, unequal learning across classes or sub-populations translates directly into unequal quality or memorization in the generated samples. Recent empirical work has shown that diffusion models can exacerbate such disparities even on datasets that appear balanced at the class level [41, 51, 61]. The mechanisms behind these effects, however, remain poorly understood.

A growing body of theory has begun to characterize the training dynamics of score-based diffusion models. On finite datasets, sufficiently expressive models trained by score matching are naturally

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

driven toward exact reproduction of the training samples, making memorization the asymptotic outcome of training [7]. The key dynamical phenomenon, however, is that this regime is not necessarily reached immediately: diffusion models can first pass through an initial phase of generalization before eventually memorizing the training set [2, 4, 6, 27]. In overparameterized models, this separation between generalization and memorization is what allows early stopping to act as an implicit regularizer [8, 15, 16]: there is a window during which the model has learned the distribution but has not yet memorized the training set.

Existing analyses of this timescale separation, however, largely characterize learning at the level of the full dataset, implicitly assuming that all sub-populations are structurally identical. Real datasets are not homogeneous: classes can differ substantially in intrinsic variance, geometric organization, and sampling frequency [37, 41, 61, 62]. Under these conditions, there is no reason to expect a single global schedule for learning or memorization. Different sub-populations may be learned at different stages of training, so that a stopping time that still regularizes one class may already leave another overfit or insufficiently learned.

In this work, we develop a high-dimensional theoretical framework for this class-aware setting by studying denoising score matching on Gaussian-mixture data with a random-features model. Our research questions are: *In what order are different classes generalized and memorized during training? How do variance, centroid geometry, and sampling imbalance shape that ordering?*

The main contributions of this work are the following:

- We characterize the effect of class imbalance on backward diffusion via class-specific inference times (see Result 1). While in the weak-imbalance regime, all classes remain inferable at the speciation time for exact models [3, 7], under strong imbalance, minority classes become inferable at distinct lower diffusion times.
- We identify the hierarchy governing class-specific learning dynamics. Class variance is the primary determinant of the order in which classes are generalized and memorized, consistently favoring higher-variance classes, while centroid geometry and sampling ratio play secondary roles. Classes with smaller-norm centroids are favored (see Result 2) and sampling imbalance acts as a modulating factor that can reverse this ordering in certain regimes (see Result 3).
- We find strong qualitative agreement between U-Net diffusion models trained on Fashion MNIST and Random Feature models trained on a Gaussian mixture parametrized by class-wise means and variances estimated from the data (see Result 4). This agreement supports, on the one hand, the effectiveness of Gaussian score approximations for diffusion models [63], and on the other, our theoretical predictions on the roles of variance and sampling imbalance in shaping learning dynamics.

A detailed discussion positioning this work within the related literature is deferred to Appendix B.

2 Problem Formulation

Score-based diffusion models transport a target data distribution $\mathcal{P}_0(\boldsymbol{x})$ to a standard N-dimensional normal distribution $\mathcal{N}(\boldsymbol{x}; \mathbf{0}, \mathbf{I}_N)$ through a forward process. Data generation is then achieved by reversing this trajectory [5, 53]. The forward and backward stochastic differential equations are respectively defined as:

$$d\boldsymbol{x}(t) = -\boldsymbol{x}dt + \sqrt{2}\, d\boldsymbol{B}(t) \qquad (1) \qquad\qquad d\boldsymbol{x}(t) = -(\boldsymbol{x} + 2\mathbf{s}(\boldsymbol{x}, t))dt + \sqrt{2}\, d\boldsymbol{B}(t) \qquad (2)$$

where $\boldsymbol{B}(t)$ is the standard Wiener process and $\mathbf{s}(\boldsymbol{x}, t) = \nabla_x \log \mathcal{P}_t(\boldsymbol{x})$ denotes the score function. The forward process induces a Gaussian transition kernel $\mathcal{P}_t(\boldsymbol{x}|\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{x}_0 e^{-t}, \Delta_t \mathbf{I}_N)$, where $\Delta_t = 1 - e^{-2t}$ is the noise variance at time t.

In practice, the exact score is inaccessible and is approximated by a model $\boldsymbol{s}(\boldsymbol{x}, t|\boldsymbol{\Theta})$—in our case, a Random Feature (RF) model [17, 44]. The parameters $\boldsymbol{\Theta}$ are learned by minimizing the score matching risk [22, 53] over a dataset $\boldsymbol{X} = \{\boldsymbol{x}_\nu\}_{\nu=1}^M$:

$$\mathcal{E}_{train}\left(\boldsymbol{X}, t|\boldsymbol{\Theta}\right) = \frac{1}{NM} \sum_{\nu=1}^{M} \mathbb{E}_{\boldsymbol{\xi}\sim\mathcal{N}(\mathbf{0},\mathbf{I}_N)} \left[\left\| \sqrt{\Delta_t}\, \boldsymbol{s}(\boldsymbol{x}_\nu(t), t|\boldsymbol{\Theta}) + \boldsymbol{\xi} \right\|^2 \right], \qquad (3)$$

where $\boldsymbol{x}_\nu(t) = \boldsymbol{x}_\nu e^{-t} + \sqrt{\Delta_t}\boldsymbol{\xi}$ represents the noisy samples at diffusion time t. The performance is evaluated on fresh samples $\boldsymbol{x} \sim \mathcal{P}_0$ through the test error (below $\boldsymbol{x}(t) = \boldsymbol{x}e^{-t} + \sqrt{\Delta_t}\boldsymbol{\xi}$)

$$\mathcal{E}_{test}\left(\mathcal{P}_0, t|\boldsymbol{\Theta}\right) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{P}_0,\boldsymbol{\xi}\sim\mathcal{N}(\mathbf{0},\mathbf{I}_N)}\left[\left\|\sqrt{\Delta_t}\,\boldsymbol{s}(\boldsymbol{x}(t), t|\boldsymbol{\Theta}) + \boldsymbol{\xi}\right\|^2\right], \tag{4}$$

or equivalently, through the score MSE in Appendix J.

Score Parameterization. We model the score function using a RF model with P hidden neurons:

$$\boldsymbol{s}(\boldsymbol{x}|\mathbf{A}, \mathbf{W}) = \mathbf{A}\,\phi\left(\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{x}\right), \tag{5}$$

where $\mathbf{W} \in \mathbb{R}^{P\times N}$ is a fixed random projection matrix with i.i.d. standard normal entries $W_{ip} \sim \mathcal{N}(0,1)$, and $\mathbf{A} \in \mathbb{R}^{N\times P}$ is the learnable weight matrix. The non-linear activation function is $\phi(\cdot) = \tanh$ and it is applied element-wise to the pre-activations. The theoretical framework we develop in the following applies to any activation function that admits a Hermite expansion. In Appendix G, we explore the role of the activation function.

Data Distribution. We assume the training dataset $\boldsymbol{X} = \{\boldsymbol{x}_\nu\}_{\nu=1}^M$ is sampled from a Gaussian Mixture Model (GMM) with C components:

$$\mathcal{P}_0\left(\boldsymbol{x}\right) = \sum_{c=1}^{C} b_c\,\mathcal{N}(\boldsymbol{x}; \boldsymbol{m}_c, \sigma_c^2\mathbf{I}_N), \tag{6}$$

where $\mathbf{m}_c$ are class centroids, σ_c^2 are class-conditional variances, and $b_c \geq 0$ are mixing proportions such that $\sum_{c=1}^C b_c = 1$. Unless stated otherwise, we assume $\forall\ c,\ ||\boldsymbol{m}_c|| = \mathcal{O}(\sqrt{N})$, $\forall\ c \neq c',\ ||\boldsymbol{m}_c - \boldsymbol{m}_{c'}|| = \mathcal{O}(\sqrt{N})$ and $\sigma_c^2 \simeq \frac{||\boldsymbol{m}_c||^2}{N} = \mathcal{O}(1)$.

Training Dynamics. Under Mean Squared Error minimization, the training dynamics of RF models are exactly solvable. Indeed, the gradient flow equation under the training loss (3) is a linear ODE for the readout weights $\mathbf{A}$. Assuming an initialization of $\mathbf{A}(0) = \mathbf{0}$, the exact solution is given by:

$$\mathbf{A}(\tau) = -\frac{1}{\sqrt{\Delta_t}}\mathbf{V}^\intercal\mathbf{U}^{-1}\left(\mathbf{I}_P - e^{-2\mathbf{U}\,\tau}\right); \tag{7}$$

where τ stands for the training time and $\mathbf{U}$ and $\mathbf{V}$ denote the feature-feature and feature-noise empirical covariance matrices:

$$\mathbf{U} = \sum_{c=1}^{C} b_c\mathbf{U}_c, \qquad \mathbf{U}_c = \frac{1}{M_c}\sum_{\nu\in I_c}\mathbb{E}_{\boldsymbol{\xi}}\left[\phi\left(\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{x}_\nu(t)\right)\phi\left(\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{x}_\nu(t)\right)^\intercal\right], \tag{8}$$

$$\mathbf{V} = \sum_{c=1}^{C} b_c\mathbf{V}_c, \qquad \mathbf{V}_c = \frac{1}{M_c}\sum_{\nu\in I_c}\mathbb{E}_{\boldsymbol{\xi}}\left[\phi\left(\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{x}_\nu(t)\right)\boldsymbol{\xi}^\intercal\right], \tag{9}$$

and I_c is the index set for samples of class c. The full derivation of the solution of the dynamics is provided in Appendix C.

High-Dimensional Asymptotics and Gaussian Equivalence. The learning dynamics are governed by the eigenspectrum of $\mathbf{U}$, which dictates the convergence timescales, while the cross-covariance $\mathbf{V}$ determines the scale of the learned weights. Following [8], we analyze these training regimes using Random Matrix Theory to characterize the spectrum of $\mathbf{U}$ in the high-dimensional proportional limit where $N, P, M \to \infty$ with fixed $\mathcal{O}(1)$ ratios $\chi_p = \frac{P}{N}$ and $\chi_m = \frac{M}{N}$.

We leverage Gaussian Equivalence (GE) techniques for RF models in a mixture setting [12, 18, 19], adopting a class-conditional viewpoint (see (Appendix D)). GE allows to represent a $\mathbb{R}^{P\times P}$ correlation matrix of non-linear features with a correlation matrix of linear features, perturbed by noise, in the limit $P \to \infty$, under the following two hypothesis: i) the data distribution has sub-gaussian tails; ii) the operator norm of the data covariance matrix is upper bounded. Although in our work we only consider gaussian components for the mixture, our results can be straightforwardly generalized

to any mixture whose components satisfy the hypothesis of GE. In the high-dimensional limit, the empirical covariance $\mathbf{U}$ admits the asymptotically equivalent formulation $\mathbf{U} \simeq \sum_{c=1}^{C} b_c \mathbf{U}_c^{(gep)}$, where:

$$\mathbf{U}_c^{(\mathrm{gep})} = \frac{1}{M_c} \sum_{\nu \in I_c} \mathbf{G}_t^{(c,\nu)} \mathbf{G}_t^{(c,\nu)\intercal} + \frac{\Delta_t}{N\sigma_c^2 e^{-2t}} \widehat{\mathbf{W}}_{t;c} \widehat{\mathbf{W}}_{t;c}^{\intercal} + \mathrm{diag}\left(\boldsymbol{d}_t^{(c)}\right). \tag{10}$$

Here, $\boldsymbol{G}_t^{(c,\nu)}$ are linear functions of the samples pre-activations $\boldsymbol{u}_\nu = \frac{1}{\sqrt{N}} \mathbf{W} \boldsymbol{x}_\nu$ with additional Gaussian noise, $\widehat{\mathbf{W}}_{t;c} \in \mathbb{R}^{P \times N}$ are cluster-specific RF matrices, and $\mathrm{diag}(\boldsymbol{d}_t^{(c)})$ is a diagonal matrix containing the vector $\boldsymbol{d}_t^{(c)} \in \mathbb{R}^P$. The first term of Eq. (10) is a linearization of Eq. (8) containing information on the training set and is responsible for the onset of memorization and diffusion-independent generalization; the second and third terms, instead, are dataset-independent and represent the t-dependent part of generalization. We explain how to derive the GEP in (10) and how to single out exactly generalization and memorization terms in Eq. (10) in Appendix E.

Learning curves. Under Eq. (7), the train and test errors of Eqs.(3),(4) can be written in terms of traces of the empirical correlations in Eqs.(8),(9) and of the population matrix $\widetilde{\mathbf{U}} = \mathbb{E}_{\boldsymbol{X}}[\mathbf{U}] = \sum_{c=1}^{C} b_c \widetilde{\mathbf{U}}_c$

$$\begin{aligned} \mathcal{E}_{train}(\tau) &= 1 - \frac{1}{N} \mathrm{Tr}\, \mathbf{V}^{\intercal} \mathrm{f}_\tau(\mathbf{U})(2\mathbf{I}_\mathrm{P} - \mathbf{U}\mathrm{f}_\tau(\mathbf{U}))\mathbf{V} \\ \mathcal{E}_{test}^{(c)}(\tau) &= 1 + \frac{1}{N} \mathrm{Tr}\, \mathbf{V}^{\intercal} \mathrm{f}_\tau(\mathbf{U}) \widetilde{\mathbf{U}}_c \mathrm{f}_\tau(\mathbf{U})\mathbf{V} - \frac{2}{\mathrm{N}} \mathrm{Tr}\, \mathbf{V}^{\intercal} \mathrm{f}_\tau(\mathbf{U})\mathbf{V}. \end{aligned} \tag{11}$$

where we set $\mathrm{f}_\tau(\mathbf{U}) = \mathbf{U}^{-1}(\mathbf{I}_\mathrm{P} - \mathrm{e}^{-2\mathbf{U}\tau})$ and decomposed the test error of Eq. (4) into class-conditional components $\mathcal{E}_{test}(\tau) = \sum_c b_c \mathcal{E}_{test}^{(c)}(\tau)$. The formulas in (11) are exact finite-dimensional $N, P, M < \infty$ learning curves for a given realization of the network and the training dataset encoded in the $\mathbf{U}, \mathbf{V}, \{\widetilde{\mathbf{U}}_c\}$; with these matrices replaced by their GE representations in Appendix E, they constitute our *semi-analytical* theoretical prediction for the dynamics (more details in Appendix I). In the high-dimensional proportional limit described in the previous paragraph, the traces in Eq. (11) concentrate around their mean: we computed the asymptotic expression for the train error in Appendix I, leaving the expressions for the test errors to future work.

3 Characterization of the Learning Dynamics

We characterize how the Random Feature model learns on an imbalanced, structured dataset by minimizing the score-matching empirical risk. We proceed in three steps: First, in Sec.3.1, we analyze the backward diffusion process to identify the class-specific inference times, formalizing when distinct data structures become inferable. Next, in Sec.3.2, we detail how these structural properties are learned during training, leveraging the eigenspectrum of the feature covariance to define global timescales for generalization and memorization. Finally, in Sec.3.2.1, we disaggregate these global phases to examine class-specific learning trajectories, demonstrating how the interplay between class variance, centroid norms, and sampling imbalance dictates the exact order in which classes are generalized and memorized.

3.1 Class Inference in Backward Diffusion

Prior work characterizes the *speciation transition* as the diffusion time at which the generative sampling process begins to resolve specific data modes [3, 7, 45]. Such definitions are model-agnostic, characterizing mode resolution only in terms of the geometric properties of data during backward diffusion. In our framework, we ask the following question: *when will a learning model infer different classes of a dataset during backward diffusion?* We define the *class-inference time* as the point along the backward diffusion trajectory where a class centroid becomes inferable through the learning of the score function.

To precisely characterize these times, we analyze the spectrum of the Gaussian equivalent feature covariance $\mathbf{U}_{\mathrm{gep}}$. Expanding Eq. (10) in the large-time regime (small e^{-t}), we obtain:

$$\mathbf{U}_{\mathrm{gep}} = \frac{\gamma^2}{N} \mathbf{W}\mathbf{W}^{\intercal} + (\widetilde{\beta} - \gamma^2)\mathbf{I}_P + \gamma^2 e^{-2t} \sum_{c=1}^{C} b_c \boldsymbol{\mu}_c \boldsymbol{\mu}_c^{\intercal} + \mathcal{O}(e^{-4t}), \tag{12}$$

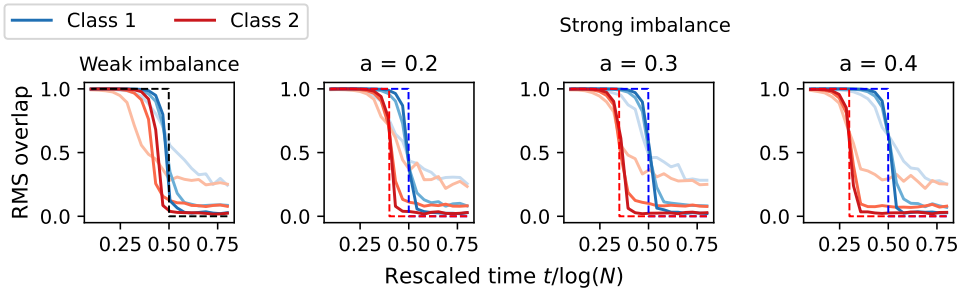


Figure 1: **Strong imbalance induces separation in inference times.** Root mean square (RMS) overlaps between the projected centroids $\boldsymbol{\mu}_c$ (Eq. (12)) and the top principal components $(\boldsymbol{\psi}_1, \boldsymbol{\psi}_2)$ of $\mathbf{U}_{\text{gep}}$, defined as $\text{RMS}_c \equiv \text{Mean}(\sum_{i=1}^{2}(\boldsymbol{\mu}_c \cdot \boldsymbol{\psi}_i)^2/\|\boldsymbol{\mu}_c\|^2)^{1/2}$, as a function of rescaled diffusion time $\widetilde{t} \equiv t/\log N$. *Left* panel corresponds to weak imbalance ($b_1 = 0.7, b_2 = 0.3$, Eq. (14)), while the remaining panels illustrate instances of strong imbalance ($b_1 = 1 - N^{-a}$, $b_2 = N^{-a}$, Eq. (15)). Dashed lines denote asymptotic theoretical predictions from Eqs.(14),(15); continuous lines represent finite-size numerical simulations for $N \in \{50, 500, 5000\}$ and $P \in \{100, 1000, 10000\}$, with increasing system size indicated by darker colors. Simulations show convergence toward the predicted asymptotic limits. Parameter values: $\sigma_1^2 = \sigma_2^2 = 0.5$, $\|\boldsymbol{m}_1\| = \|\boldsymbol{m}_2\| = \sqrt{N}$, $\boldsymbol{m}_1 \cdot \boldsymbol{m}_2 = 0$, $M = 2000$, $N_{runs} = 50$ for $P = 100, 1000$ and $N_{runs} = 10$ for $P = 10000$.

where $\widetilde{\beta} = \mathbb{E}_{z\sim\mathcal{N}(0,1)}\left[\phi^2(z)\right]$, $\gamma = \mathbb{E}_{z\sim\mathcal{N}(0,1)}\left[z\phi(z)\right]$, and $\boldsymbol{\mu}_c = \frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{m}_c$ are the projected class centroids. The derivation of (12) can be found in Appendix F.

Eq. (12) shows that, in the high-dimensional limit ($N \to \infty$), a class becomes inferable when its corresponding rank-1 perturbation $\boldsymbol{\mu}_c\boldsymbol{\mu}_c^\intercal$ produces an isolated eigenvalue that escapes the continuous bulk of the matrix spectrum. This spectral phase transition occurs when the perturbation norm is non-vanishing, yielding the condition:

$$\gamma^2 e^{-2t}\|\boldsymbol{\mu}_c\|^2 b_c = \mathcal{O}(1). \tag{13}$$

Assuming the initial class centroids are extensive, i.e. $\|\boldsymbol{m}_c\|^2 = \text{m}_c^2 N$, the required diffusion time to allow inference scales as $t = \mathcal{O}(\log N)$. By varying the asymptotic scaling of the sampling weights b_c, we can evaluate how class structure dictates the temporal emergence of features:

Result 1 (Class-Specific Inference Times)**.** *Let b_c denote the sampling weight of class c, assume the validity of the GE reduction. The inference time $t_I^{(c)}$ is governed by two distinct data regimes:*

- **Weak Imbalance:** *When all classes are proportionally represented ($b_c = \mathcal{O}(1)$), class imbalance has no leading-order effect on the backward dynamics. All classes are inferred simultaneously at the speciation time t_s:*

$$t_I^{(c)} = \frac{1}{2}\log N + \mathcal{O}(1) \equiv t_s. \tag{14}$$

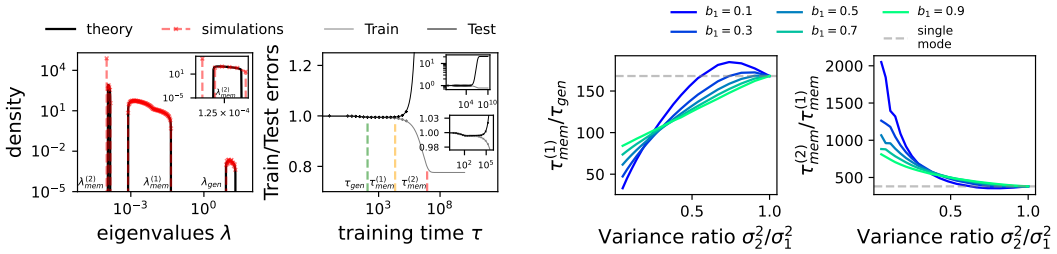
- **Strong Imbalance:** *When a class is severely undersampled ($b_c = \mathcal{O}(N^{-a_c})$ for $0 < a_c < 1$), its transition is delayed. The system exhibits a hierarchy of class-dependent inference times:*

$$t_I^{(c)} = \frac{1 - a_c}{2}\log N + \mathcal{O}(1) \leq t_s. \tag{15}$$

Result 1 shows that while for mild data imbalance class inference corresponds asymptotically to class speciation, strong imbalance breaks this symmetry. The generative model must infer the dataset structure sequentially, prioritizing overrepresented classes and suppressing the emergence of minority classes until much later in the backward process (i.e., closer to $t = 0$). Fig.1 validates these theoretical predictions against finite-size numerical simulations for $C = 2$, demonstrating good agreement as $N, P \to \infty$.

Note that an equivalent hierarchy of inference times emerges even under balanced and weakly imbalanced sampling ($b_c = \mathcal{O}(1)$) if the classes have subleading geometric properties. Specifically, if a centroid is subextensive such that $\|\boldsymbol{m}_c\|^2 = \mathcal{O}(N^{1-a_c})$, Eq. (13) trivially dictates that its inference time will be equivalently delayed by $\frac{1-a_c}{2}\log N$.

3.2 Timescale Separation: Generalization vs. Memorization



a Spectral density and error dynamics. *Left:* Theoretical spectral density of $\mathbf{U}$ (solid line) versus numerical histograms, highlighting the three critical eigenvalues λ_{gen}, $\lambda^{(1)}_{mem}$, and $\lambda^{(2)}_{mem}$ bounding the primary eigenvalue bulks. *Right:* Train and test losses evaluated over training time τ. Analytical predictions (solid lines) tightly track numerical gradient descent averages (dots). The vertical dashed lines denote the timescales corresponding to the critical eigenvalues.

b Generalization and memorization windows. Theoretical widths of the generalization window w_g *(Left)* and memorization window w_m *(Right)* as a function of the variance ratio $v = \sigma_2^2/\sigma_1^2$, plotted for varying mixing fraction b_1. The dashed baseline represents the standard unimodal setting. Reducing the variance ratio extends memorization but induces a non-monotonic contraction in the generalization window, an effect amplified by undersampling the high-variance class.

Figure 2: **Strong class diversity can reduce the generalization window**. Parameters for theory: $\chi_p = 60.0$, $\chi_m = 30.0$, $b_1 = b_2 = 0.5$, $\sigma_1^2 = 0.5, \sigma_2^2 = 0.25$, $\|\boldsymbol{m}_1\| = \|\boldsymbol{m}_2\| = 0$, $t = 0.001$. Parameters for finite-size simulations: $N = 100$, $P = 6000$, $M = 3000$, $N_{epoch} = 2 \times 10^6$, learning rate $\eta = 5 \times 10^{-5} N/\Delta_t$, $N^{spectrum}_{run} = 10$, $N^{train}_{run} = 20$. Training time in (**a**) is rescaled as $\tau \to \tau/\widetilde{\eta}$, where $\widetilde{\eta} = \eta \Delta_t / N$.

Having established how data structure emerges across diffusion time (t), we now analyze the network's learning dynamics over training time (τ). Specifically, we investigate how the eigenspectrum of the feature covariance matrix $\mathbf{U}$ dictates the global timescales of generalization and memorization during gradient descent.

For clarity, we restrict our focus to a binary Gaussian mixture ($C = 2$) evaluated at small diffusion times ($t \ll 1$). We quantify class diversity via the variance ratio $v \equiv \sigma_2^2/\sigma_1^2$, treating it as a structural perturbation to the unimodal baseline ($v = 1$). Leveraging the heuristic Replica Method from Statistical Physics (detailed in Appendix H), we derive the spectral density of $\mathbf{U}$ in the asymptotic proportional limit described in 2. As shown in the left panel of Fig.2**a**, the spectrum exhibits distinct bulk regions whose extremal eigenvalues govern distinct phases of the training trajectory. In particular, the lower edge of the rightmost bulk λ_{gen}, the upper edge of the central bulk $\lambda^{(1)}_{mem}$, and the lower edge of the leftmost bulk $\lambda^{(2)}_{mem}$, identify three interpretable timescales:

- $\tau_{gen} = (2\lambda_{gen})^{-1}$: onset of the generalization phase.
- $\tau^{(1)}_{mem} = (2\lambda^{(1)}_{mem})^{-1}$: end of generalization and onset of data memorization.
- $\tau^{(2)}_{mem} = (2\lambda^{(2)}_{mem})^{-1}$: completion of the memorization/overfitting phase.

In contrast to the first two timescales, the latter depends on diffusion time, with $\tau^{(2)}_{mem} \propto 1/t$ for $t \ll 1$ (see Appendix H.3.1), indicating that the final memorization stage occurs only at asymptotically long training times. Based on a simple counting argument on the eigenvalue density (see Appendix H.3.1), the training phase corresponding to the central bulk can be associated with the creation of basins of attraction around each training sample, with widths set by the forward noise variance $\Delta_t \approx 2t$. The remaining training phase, therefore, corresponds to small refinements within these narrow basins. Although this final regime is theoretically relevant to characterize the full memorization window, it is unlikely to be observed empirically: finite learning rates effectively bypass the associated small-eigenvalue directions, and nearest-neighbor metrics will typically deem the data fully memorized before $\tau^{(2)}_{mem}$ is reached.

As illustrated in Fig. 2**a**, the theory characterizes the spectrum and corresponding spectral edges map to the macroscopic transitions in the empirical train and test losses. To evaluate the persistence of these learning phases, we define the generalization window $w_g = \tau^{(1)}_{mem}/\tau_{gen}$ and the memorization

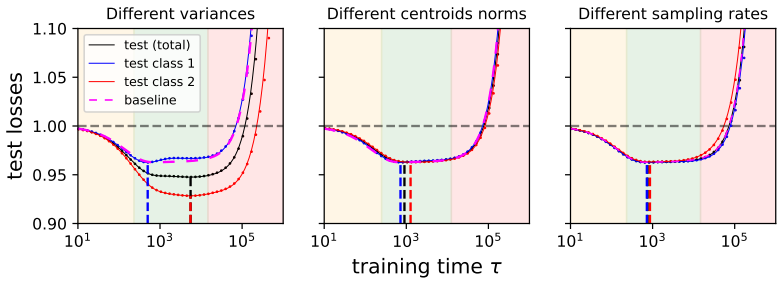


Figure 3: **Class heterogeneity induces class-specific timescales.** Class-conditional test errors (Eq. 11) evaluated over training time across three structural ablations relative to a perfectly symmetric baseline ($\sigma_1^2 = \sigma_2^2 = 0.5$, $\|\boldsymbol{m}_1\| = \|\boldsymbol{m}_2\| = \sqrt{N}$, $b_1 = b_2 = 0.5$, $\boldsymbol{m}_1 \cdot \boldsymbol{m}_2 = 0$). The perturbations are: variance disparity ($\sigma_1^2 = 0.5, \sigma_2^2 = 0.25$; *left*), centroid norm disparity ($\|\boldsymbol{m}_1\| = \sqrt{N}$, $\|\boldsymbol{m}_2\| = \frac{3}{2}\sqrt{N}$; *center*), and sampling imbalance ($b_1 = 0.8, b_2 = 0.2$; *right*). Solid lines denote theoretical predictions (Appendix I), dots represent empirical averages from gradient descent simulations and vertical dashed lines refer to the positions of the $\tau_{gen}^{(c)}, \quad c = 1, 2$. Background colors characterize the training windows theoretically derived from the spectrum of $\mathbf{U}_{\text{gep}}$: early (orange), generalization (green), and memorization (red). Other parameters: $\chi_p = 60.0$, $\chi_m = 30.0$, $t = 0.01$, $N = 100$, $N_{epoch} = 2 \times 10^6$, learning rate $\eta = 5 \times 10^{-5} N/\Delta_t$, $N_{run}^{train} = 50$.

window $w_m = \tau_{mem}^{(2)}/\tau_{mem}^{(1)}$. Fig. 2**b** plots these windows as a function of the variance ratio v. We observe that strong class diversity (i.e. v deviating significantly from 1) reliably contracts the generalization window relative to the unimodal baseline, while strictly expanding the memorization window. In regimes of high class diversity ($v \ll 1$), imbalance modulates the width of w_g: imbalance toward the high-variance class mitigates the structural effects of class diversity, whereas imbalance toward the low-variance class amplifies the contraction.

3.2.1 Class-Specific Learning Dynamics

To quantify the performance on each class in the dataset, we consider the semi-analytic class-specific test errors in Eq. (11). By tracking the class-specific loss trajectories, we formalize two critical times in the training process. The optimal generalization time, $\tau_g^{(c)}$, is defined as the training time that minimizes the class-conditional test error in Eq. (11). Following this initial generalization phase, the empirical risk minimization on a finite dataset inevitably induces overfitting. We denote the onset of pure memorization, $\tau_m^{(c)} > \tau_g^{(c)}$, as the threshold where this overfitting causes the test error to degrade back to 1.[1] Formally:

$$\tau_g^{(c)} = \underset{\tau}{\text{argmin}}\ \mathcal{E}_{test}^{(c)}(\tau), \qquad \mathcal{E}_{test}^{(c)}(\tau_m^{(c)}) = 1. \tag{16}$$

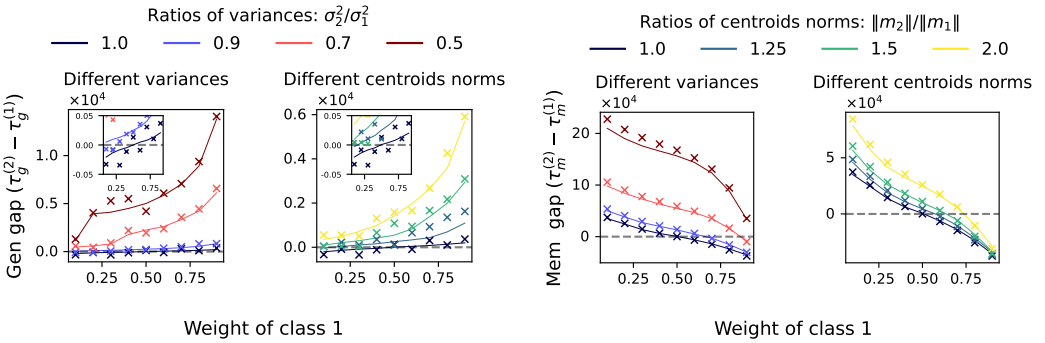
Fig.3 shows the test error under three structural ablations relative to a balanced baseline with orthogonal, equal-norm centroids: doubling the variance of class 2 (left), multiplying the centroid norm of class 2 by $3/2$ (center), and skewing the sampling weights (right). These perturbations reveal a hierarchy of influence:

Result 2 (Hierarchy of Class Learning)**.** *Suppose the per-class generalization and memorization times are measured from the semi-analytical curves of Eq.*(11)*, under the GE reduction and the assumptions in Section 2. The learning timescales are governed by the following factors:*

- **Primary: Variance**. *High-variance classes present smoother and larger score targets, leading to significantly faster generalization and memorization compared to low-variance components.*
- **Secondary: Centroid Geometry** and **Sampling Imbalance**. *Classes with smaller-norm centroids are prioritized over those with larger-norm centroids. Strong imbalance acts as a modulator capable of inverting the learning order established by variance and geometry.*

[1] Because the readout weights are initialized at zero ($\mathbf{A}(0) = \mathbf{0}$, see Sec.2), the network outputs zero at initialization. Therefore, the initial test error is trivially equal to the variance of the standard Gaussian noise: $\mathcal{E}_{test}^{(c)}(0) = 1$.

For completeness, Appendix J provides analogous results evaluated on the exact score-matching Mean Squared Error.



a Generalization gaps. Gap between optimal generalization training time of the two classes against the sampling weight of class 1 (Eq. (16)). Panels isolate the impact of variance disparity (*Left*, with inset zoom) and the centroid norm disparity (*Right*) introduced in Fig. 3.

b Memorization gaps. Gap between memorization times of the two classes against the sampling weight of class 1 (Eq. (16)). As in **a**, the panels show the variance disparity setting (*Left*) and the centroid norm disparity setting (*Right*).

Figure 4: **Imbalance can reverse the order of learning for classes**. In both subfigures, solid lines represent theoretical predictions derived in Appendix I, while crosses represent empirical average gaps from gradient descent simulations. Parameters: $\chi_p = 60.0$, $\chi_m = 30.0$, $t = 0.01$, $N = 100$, $N_{epoch} = 2 \times 10^6$, learning rate $\eta = 5 \times 10^{-5} N/\Delta_t$, $N_{run}^{train} = 50$.

As illustrated in Fig. 4, class imbalance acts as a modulator capable of inverting the learning order observed in the balanced setting. By studying the memorization and generalization gaps between classes as functions of the mixing proportion b_1, we identify a fundamental trade-off:

Result 3 (The Generalization-Memorization Trade-off)**.** *Sample imbalance impacts generalization and memorization in opposite directions:*

1. **Scarcity delays generalization:** *Undersampling the high-variance class forces the model to generalize the low-variance class first.*
2. **Scarcity accelerates memorization:** *Conversely, oversampling the high-variance class deprives the low-variance class of data, causing it to be memorized first.*

Consequently, adjusting sampling weights to close the generalization gap between classes inevitably widens the memorization gap, and vice versa.

4 Numerical Experiments on Fashion MNIST

Model, Training, and Generation. A standard DDPM [22] was employed on the Fashion MNIST dataset [65], with a U-Net [46] serving as the backbone. Each model is trained on a binary subset of Fashion MNIST, pairing the Sneaker class—which exhibits the lowest pixel-wise variance—with the remaining nine classes to yield nine distinct class pairs. Each pair is trained at three Sneaker proportions (25%, 50%, 75%) on 8,000 images. To capture both early and late training dynamics, 32 log-spaced checkpoints are saved per run. At each checkpoint, the DDPM ancestral sampler [22] is used to generate 500 samples per pair. Further implementation details are deferred to Appendix L.1.

Evaluation. Following [8, 20, 66], a generated sample $\boldsymbol{x}_c$ is considered memorized if the distance ratio to its nearest training neighbors nn_1, nn_2 satisfies $\frac{\|\boldsymbol{x}_c - \mathrm{nn}_1(\boldsymbol{x}_c)\|_2}{\|\boldsymbol{x}_c - \mathrm{nn}_2(\boldsymbol{x}_c)\|_2} < \frac{1}{3}$. The quantity $f_{\mathrm{mem}}^c(t)$ is the expected memorization fraction under the data distribution at diffusion time t. We define the threshold-crossing time $t_{\mathrm{mem}}^c = \inf\{t : f_{\mathrm{mem}}^c(t) \geq 1/3\}$, yielding the Sneaker-partner memorization gap $\Delta t_{\mathrm{mem}}(c_p) = t_{\mathrm{mem}}^{\mathrm{snk}} - t_{\mathrm{mem}}^{c_p}$. Samples are assigned to classes using dedicated binary ResNet-18 classifiers [21] (macro F1 $> 96.0\%$, balanced accuracy $> 97.3\%$).

Results and Discussion. We first verify whether the structural hierarchy predicted by Result 2 and shown in Fig.3 is observed in Fashion MNIST DDPMs. As shown in Fig.5**a**, in the Sneaker-Bag pairs,

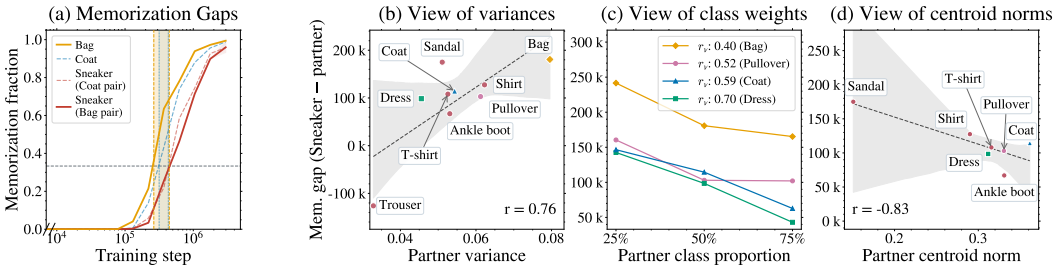


Figure 5: **DDPM memorization gaps on Fashion MNIST.** (**a**) **Per-class memorization fractions for Sneaker-Coat and Sneaker-Bag DDPMs.** The horizontal dashed line marks the threshold $f^c_{\text{mem}}(t) = 1/3$ used to define t^c_{mem}; vertical shaded regions show the resulting Sneaker-partner time gap $\Delta t_{\text{mem}}(c_p)$. Within each pair, the higher-variance partner class is memorized before Sneaker. Across pairs, the Bag-Sneaker DDPM exhibits a larger memorization gap than the Coat-Sneaker DDPM. (**b**) **Memorization gap versus the empirical mean pixel variance of the balanced partner class.** The dashed line is a linear fit, and the gray band is a 95% bootstrap confidence interval obtained by resampling class points and refitting. (**c**) **Memorization gap across partner mixing proportions b_{c_p}.** The legend reports $r_v = \widehat{\sigma}^2_{\text{snk}}/\widehat{\sigma}^2_{c_p}$. The selected classes are ordered in roughly 0.1 increments of r_v; apart from Bag, they have comparable empirical variances. The trend is consistent with the one in Fig. 4**b**. (**d**) **Memorization gap versus partner centroid norm.** To isolate centroid effects, we exclude relatively extreme variance cases, computed on the balanced subset; the fit and bootstrap confidence interval are computed as in (b).

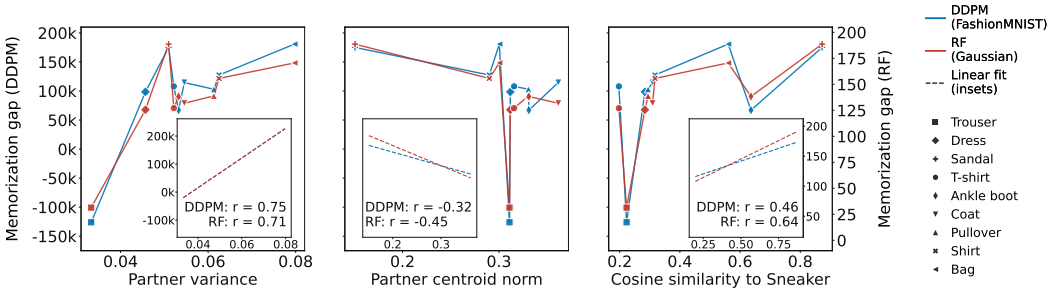


Figure 6: **DDPM–RF comparison across three descriptors.** For each Sneaker-partner pair (c_{snk}, c_p), we compare the DDPM memorization gap on Fashion MNIST with the corresponding RF memorization gap on Gaussian synthetic data. The RF model uses pair descriptors extracted from the data: partner variance, normalized partner centroid norm, and centroid cosine similarity to Sneaker. The panels plot both gaps as functions of each descriptor; connecting lines are visual guides only. Insets report linear fits and Pearson correlations in the same raw units as the outer panels.

the higher-variance class attains the memorization threshold before its lower-variance counterpart. This relationship between the largest and smallest variance classes is also corroborated on CIFAR in Appendix L.2. Moreover, in Fig.5**b**, the memorization time gap $\Delta t_{\text{mem}}(c_p)$ is positively correlated with the empirical mean pixel variance of different partner classes. Additionally, imbalance acts as a modulator: Fig.5**c** illustrates that increasing the partner mixing proportion b_{c_p} generally reduces the memorization gap. The curves in Fig.5**c** maintain a vertical ordering consistent with the variance ratio $r_v = \widehat{\sigma}^2_{\text{snk}}/\widehat{\sigma}^2_{c_p}$, indicating that variance remains a dominant factor across the tested imbalance regimes. Finally, Fig.5**d** isolates the effect of centroid geometry by excluding extreme-variance cases. It confirms the theoretical prediction that partner classes with a lower centroid norm relative to the Sneaker class exhibit larger memorization gaps, an observation consistent with the comparatively large gap recorded for the Sneaker-Sandal pair.

While previous sections isolated specific data properties on synthetic data, in these numerical experiments we observe their combined effect. Additionally, each class has its own non-null overlap with the reference Sneaker class, reflected by the cosine similarity. To generalize the theory to this case, we construct a synthetic Gaussian mixture model (GMM) using the centroid norms, variances, and cosine similarities extracted from Fashion MNIST.

Result 4 (Consistency between RF learning and Real Architectures)**.** *Using only these pair-level statistics, the RF model evaluated on Gaussian synthetic data qualitatively matches the memorization-gap trends of the full DDPM (Fig.6). Despite operating on distinct gap scales, the models share similar descriptor-dependent patterns:*

- **Qualitative agreement:** *Both models exhibit larger gaps for partner classes with larger variances and smaller centroid norms, matching the theoretical predictions. This similarity emerges despite comparing distinct architectures (RF vs. U-Net) and datasets (synthetic GMM vs. Fashion MNIST).*
- **Pairwise structure:** *The RF model captures specific geometric relationships precisely. For instance, when the pairs have similar variances and cosine similarities, the one with a smaller centroid norm results in a larger memorization gap in both models (e.g., see the Shirt pair compared to the Pullover pair). Likewise, a partner class exhibiting moderate variance and a very small centroid norm results in a large memorization gap in both settings (e.g., see the Sandal pair).*

These results indicate that a Gaussian structural representation captures the leading-order effects of the memorization hierarchy in real image data. While first- and second-order moments do not fully resolve the complexity of memorization—as evidenced by residual discrepancies—they successfully bridge the high-dimensional theory with empirical practice.

5 Conclusions

We developed a high-dimensional analytical framework for class-aware learning dynamics in score-based diffusion models, characterizing the interplay of data structure and sampling imbalance. We showed that strong imbalance induces distinct class-specific inference times, and that during score learning—close to sampling time—the statistical properties of the data, especially class variances, are the leading factors governing the order in which classes are learned. Centroid geometry and sampling imbalance play secondary roles, with the latter acting as a modulator that can reverse this ordering. We further found that U-Net DDPMs trained on Fashion MNIST qualitatively track these predictions when an RF model is parametrized by empirical class moments, supporting the relevance of Gaussian score approximations to real architectures. A natural practical implication of our results is that a single early-stopping time need not regularize all classes equally under heterogeneous data. Since classes with higher variance or smaller centroid norm are learned earlier, reweighting or rebalancing toward lower-signal classes may offer a way to mitigate per-class disparities. We leave a systematic study of such interventions to future work.

Our analysis is limited in two main respects. First, our presentation focused on binary Gaussian mixtures with isotropic covariances, even though the framework naturally extends to multi-class and non-isotropic settings. Despite the simplicity of such distributions, influent works about the theory of generative diffusion [7, 45] already relied on Gaussian distributions. Their motivation is, together with a higher interpretability of the model, also the fact that at high and intermediate noise levels diffusion models effectively behave like Gaussian mixtures [63]. The same hypothesis seems transposable to training time, as supported by our experimental analysis in Section 4, where we find a remarkable agreement between measures performed on U-Nets and Gaussian mixture models across learning. Furthermore, our analysis is fundamentally restricted to capturing first- and second-order structures. This leaves open the question of how more complex heterogeneities affect generative performance. For example, dataset outliers may act as strong attractors in diffusion dynamics [25], and real data classes often have internal multimodal structure that cannot be reduced to a single Gaussian component. Moreover, future work could study data with power-law spectra. Such covariance eigenvalue distributions are common in natural images [48] and they are suspected to bring interesting implications for training diffusion models [37]. Since weaker intra-class modes can also be suppressed by conditional sampling and guidance [26], extending the theory to more realistic class structure and conditional settings would be an important next step.

Acknowledgments and Disclosure of Fundings

The authors thank Giulio Biroli, Sebastian Goldt, and Bruno Loureiro for their precious feedback. F.N., C.M., and S.S.M. were supported by the Wallenberg AI, Autonomous Systems, and Software Program (WASP). The computations were enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS) at Alvis and Tetralith (Chalmers Centre for Computational Science and Engineering, C3SE) partially funded by the Swedish Research Council through grant agreement no. 2022-06725, under project NAISS 2024/22-1082.

References

[1] B. Achilli et al. "Losing dimensions: Geometric memorization in generative diffusion". In: *arXiv:2410.08727* (2024).

[2] B. Achilli et al. "Memorization and generalization in generative diffusion under the manifold hypothesis". In: *Journal of Statistical Mechanics: Theory and Experiment* 2025.7 (2025), p. 073401.

[3] B. Achilli et al. "Theory of speciation transitions in diffusion models with general class structure". In: *Journal of Statistical Mechanics: Theory and Experiment* 2026.4 (2026), p. 043304.

[4] L. Ambrogioni. "The statistical thermodynamics of generative diffusion models: Phase transitions, symmetry breaking and critical instability". In: *arXiv:2310.17467* (2024).

[5] B. D. Anderson. "Reverse-time diffusion equation models". In: *Stochastic Processes and their Applications* 12.3 (1982), pp. 313–326.

[6] G. Biroli and M. Mézard. "Generative diffusion in very large dimensions". In: *Journal of Statistical Mechanics: Theory and Experiment* 2023.9 (2023), p. 093402.

[7] G. Biroli et al. "Dynamical regimes of diffusion models". In: *Nature Communications* 15.1 (2024), p. 9957.

[8] T. Bonnaire et al. "Why Diffusion Models Don't Memorize: The Role of Implicit Dynamical Regularization in Training". In: *Advances in Neural Information Processing Systems* (2025).

[9] P. Charbonneau et al. *Spin glass theory and far beyond: replica symmetry breaking after 40 years*. World Scientific, 2023.

[10] H. Cui, C. Pehlevan, and Y. M. Lu. "A solvable model of learning generative diffusion: theory and insights". In: *arXiv preprint arXiv:2501.03937* (2025).

[11] H. Cui et al. "Analysis of learning a flow-based generative model from limited sample complexity". In: *arXiv preprint arXiv:2310.03575* (2023).

[12] Y. Dandi et al. "Universality laws for gaussian mixtures in generalized linear models". In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 54754–54768.

[13] J. Devlin et al. "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *Proceedings of the 2019 conference of the North American association for computational linguistics* (2019), pp. 4171–4186.

[14] S. F. Edwards and R. C. Jones. "The eigenvalue spectrum of a large symmetric random matrix". In: *Journal of Physics A: Mathematical and General* 9.10 (1976), p. 1595.

[15] A. Favero, A. Sclocchi, and M. Wyart. "Bigger Isn't Always Memorizing: Early Stopping Overparameterized Diffusion Models". In: *arXiv preprint arXiv:2505.16959* (2025).

[16] A. J. George, R. Veiga, and N. Macris. "Analysis of diffusion models for manifold data". In: *2025 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2025, pp. 1–6.

[17] A. J. George, R. Veiga, and N. Macris. "Denoising Score Matching with Random Features: Insights on Diffusion Models from Precise Learning Curves". In: *Arxiv:2502.00336* (2025).

[18] F. Gerace et al. "Generalisation error in learning with random features and the hidden manifold model". In: *International Conference on Machine Learning*. PMLR. 2020, pp. 3452–3462.

[19] S. Goldt et al. "Modeling the influence of data structure on learning in neural networks: The hidden manifold model". In: *Physical Review X* 10.4 (2020), p. 041044.

[20] X. Gu et al. "On memorization in diffusion models". In: *arXiv preprint arXiv:2310.02664* (2023).

[21] K. He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[22] J. Ho, A. Jain, and P. Abbeel. "Denoising Diffusion Probabilistic Models". In: *Advances in Neural Information Processing Systems* (2020).

[23] J. Howard and S. Ruder. "Universal language model fine-tuning for text classification". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2018, pp. 328–339.

[24] A. Jain et al. "Bias in motion: Theoretical insights into the dynamics of bias in sgd training". In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 24435–24471.

[25] D. Jeon, D. Kim, and A. No. "Understanding and mitigating memorization in generative models via sharpness of probability landscapes". In: *International Conference on Learning Representations* (2025).

[26] C. Jin, Q. Shi, and Y. Gu. "Stage-wise dynamics of classifier-free guidance in diffusion models". In: *arXiv preprint arXiv:2509.22007* (2025).

[27] Z. Kadkhodaie et al. "Generalization in diffusion models arises from geometry-adaptive harmonic representation". In: *International Conference on Learning Representations* (2024).

[28] D. P. Kingma and J. Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[29] A. Krizhevsky, G. Hinton, et al. *Learning multiple layers of features from tiny images*. Toronto, ON, Canada, 2009.

[30] A. Krogh and J. Hertz. "A simple weight decay can improve generalization". In: *Advances in neural information processing systems* 4 (1991).

[31] M. Kwon, J. Jeong, and Y. Uh. "Diffusion models already have a semantic latent space". In: *arXiv preprint arXiv:2210.10960* (2022).

[32] H. Latourelle-Vigeant et al. *Generalization, memorization, and overfitting for diffusion models trained in the lazy high-dimensional regime*. 2026. arXiv: `2608.23938 [stat.ML]`. URL: `https://arxiv.org/abs/2608.23938`.

[33] Q. Liang et al. "How diffusion models learn to factorize and compose". In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 15121–15148.

[34] I. Loshchilov and F. Hutter. "Sgdr: Stochastic gradient descent with warm restarts". In: *arXiv preprint arXiv:1608.03983* (2016).

[35] I. Loshchilov and F. Hutter. "Decoupled weight decay regularization". In: *arXiv preprint arXiv:1711.05101* (2017).

[36] V. C. Mendes et al. "A solvable high-dimensional model where nonlinear autoencoders learn structure invisible to PCA while test loss misaligns with generalization". In: *arXiv:2602.10680* (2026).

[37] C. Merger and S. Goldt. "Generalization dynamics of linear diffusion models". In: *arXiv preprint arXiv:2505.24769* (2025).

[38] C. Merger and S. Goldt. *Local Coverage Governs Memorization in Diffusion Models*. 2026. arXiv: `2606.14390 [cond-mat.dis-nn]`. URL: `https://arxiv.org/abs/2606.14390`.

[39] A. Montanari and P. Urbani. "Dynamical decoupling of generalization and overfitting in large two-layer networks". In: *arXiv preprint arXiv:2502.21269* (2025).

[40] W. Peng et al. "Diffusion Models Learn Low-Dimensional Distributions via Subspace Clustering". In: *arXiv preprint arXiv:2409.02426* (2024).

[41] M. V. Perera and V. M. Patel. "Analyzing bias in diffusion-based face generation models". In: *2023 IEEE International Joint Conference on Biometrics (IJCB)*. IEEE. 2023, pp. 1–10.

[42] F. S. Pezzicoli et al. "Class Imbalance in Anomaly Detection: Learning from an Exactly Solvable Model". In: Proceedings of Machine Learning Research (2025). Ed. by Y. Li et al., pp. 1261–1269. URL: `https://proceedings.mlr.press/v258/pezzicoli25a.html`.

[43] M. Potters and J.-P. Bouchaud. *A first course in random matrix theory: for physicists, engineers and data scientists*. Cambridge University Press, 2020.

[44] A. Rahimi and B. Recht. "Random features for large-scale kernel machines". In: *Advances in neural information processing systems* 20 (2007).

[45] G. Raya and L. Ambrogioni. "Spontaneous symmetry breaking in generative diffusion models". In: *Advances in Neural Information Processing Systems* (2023).

[46] O. Ronneberger, P. Fischer, and T. Brox. "U-net: Convolutional networks for biomedical image segmentation". In: *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.

[47] B. L. Ross et al. “A Geometric Framework for Understanding Memorization in Generative Models”. In: *ICML 2024 Next Generation of AI Safety Workshop*. 2024.

[48] D. L. Ruderman and W. Bialek. “Statistics of natural images: Scaling in the woods”. In: *Physical Review Letters* 73 (1994), p. 814.

[49] S. Sagawa et al. “An investigation of why overparameterization exacerbates spurious correlations”. In: *International Conference on Machine Learning*. PMLR. 2020, pp. 8346–8356.

[50] S. Sarao Mannelli et al. “Bias-inducing geometries: An exactly solvable data model with fairness implications”. In: *Physical Review E* 112.2 (2025), p. 025304.

[51] Y. Shi et al. “Dissecting and mitigating diffusion bias via mechanistic interpretability”. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. 2025, pp. 8192–8202.

[52] Y. Song et al. “Maximum likelihood training of score-based diffusion models”. In: *Advances in neural information processing systems* 34 (2021), pp. 1415–1428.

[53] Y. Song et al. “Score-Based Generative Modeling through Stochastic Differential Equations”. In: *International Conference on Learning Representations* (2021).

[54] N. Srivastava et al. “Dropout: a simple way to prevent neural networks from overfitting”. In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[55] J. Stanczuk et al. “Your diffusion model secretly knows the dimension of the data manifold”. In: *arXiv:2212.12611* (2023).

[56] C. Szegedy et al. “Rethinking the inception architecture for computer vision”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2818–2826.

[57] R. Urfin et al. *Double Descent and Malign Overfitting in Diffusion Models*. 2026. arXiv: 2609.26392 [cs.LG]. URL: https://arxiv.org/abs/2609.26392.

[58] R. Urfin et al. *First Learn, Then Memorize: The Spectral Bias of Diffusion Models*. 2026. arXiv: 2609.35377 [cs.LG]. URL: https://arxiv.org/abs/2609.35377.

[59] A. Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems* 30 (2017).

[60] E. Ventura et al. “Manifolds, Random Matrices and Spectral Gaps: The geometric phases of generative diffusion”. In: *International Conference on Learning Representations* (2025).

[61] J. Vice et al. “Exploring bias in over 100 text-to-image generative models”. In: *arXiv preprint arXiv:2503.08012* (2025).

[62] B. Wang and C. Pehlevan. “An analytical theory of spectral bias in the learning dynamics of diffusion models”. In: *arXiv preprint arXiv:2503.03206* (2025).

[63] B. Wang and J. J. Vastola. “The Unreasonable Effectiveness of Gaussian Score Approximation for Diffusion Models and its Applications”. In: *Transactions on Machine Learning Research* (2024).

[64] D. Weitzner et al. “The Diffusion Process as a Correlation Machine: Linear Denoising Insights”. In: *Transactions on Machine Learning Research* (2025).

[65] H. Xiao, K. Rasul, and R. Vollgraf. “Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms”. In: *arXiv preprint arXiv:1708.07747* (2017).

[66] T. Yoon et al. “Diffusion probabilistic models generalize when they fail to memorize”. In: *ICML 2023 workshop on structured probabilistic inference & generative modeling*. 2023.

Appendices

A Notation 14

B Further related works 15

Appendices for theory

C The solution of the dynamics 17

D The GEP for non-centered data 17

E The GEP of feature correlation matrices 18

E.1 The GEP of $\mathbf{U}$ 19

E.2 The GEP of $\widetilde{\mathbf{U}}$ 22

E.3 The GEP of $\mathbf{V}$ 22

E.4 The GEP of $\widetilde{\mathbf{V}}$ 23

F The matrix $\mathbf{U}$ at large t 23

G The role of the activation function 24

H Spectral equations of $\mathbf{U}$ 26

H.1 Replica computation of the resolvent function 27

H.2 Spectral equations for extensive centroids 35

H.3 Spectral equations for subextensive centroids 38

I Theoretical curves of train and test errors 42

J Score MSE 44

Appendices for experiments

K Numerical experiments on Random Features 47

K.1 Training of RF model 47

K.2 Numerical diagonalization of $\mathbf{U}_{gep}, \widetilde{\mathbf{U}}_{gep}$ 47

K.3 Solution of the spectral equations 48

L Numerical experiments on Fashion MNIST 48

L.1 Additional simulations framework details 48

L.2 Supplementary figures 49

A Notation

We summarize the nontrivial notation used throughout the paper.

For the Hadamard product, given $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^N$,

$$\boldsymbol{x} \odot \boldsymbol{y} = (x_1 y_1, \dots, x_N y_N)^\intercal. \tag{17}$$

For matrices $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{N \times N}$, the product is applied elementwise:

$$(\mathbf{X} \odot \mathbf{Y})_{ij} = X_{ij} Y_{ij}. \tag{18}$$

Notation	Meaning
x^2 vs. $x^{(2)}$	Superscripts without parentheses denote powers, while superscripts in parentheses index elements of a list or sequence.
$\boldsymbol{x}, \mathbf{X}$	Bold lowercase symbols denote vectors and bold uppercase symbols denote matrices, unless stated otherwise. For example, if $\boldsymbol{x} \in \mathbb{R}^N$, then $\boldsymbol{x} = (x_1, \dots, x_N)^\intercal$.
$\phi(\boldsymbol{x})$	For a scalar function $\phi : \mathbb{R} \to \mathbb{R}$, unless specified otherwise, $\phi(\boldsymbol{x})$ denotes elementwise application, i.e., $\phi(\boldsymbol{x}) = (\phi(x_1), \dots, \phi(x_N))^\intercal$.
$\odot$	Hadamard, or elementwise, product between vectors or matrices.
$\boldsymbol{x}^{\odot a}$	Elementwise power: $\boldsymbol{x}^{\odot a} = (x_1^a, \dots, x_N^a)^\intercal$.
$\otimes$	Kronecker product between matrices.
$\boldsymbol{x}\boldsymbol{x}^\intercal$	Outer product of a vector with itself.
$\mathbb{E}_{x\sim\mathcal{P}}[\cdot]$	Expectation with respect to $x \sim \mathcal{P}$. When the distribution is omitted, the expectation is taken over the standard normal distribution $\mathcal{N}(0, 1)$.

For matrices $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{N\times N}$, the Kronecker product is

$$\mathbf{X} \otimes \mathbf{Y} = \begin{pmatrix} X_{11}\mathbf{Y} & \cdots & X_{1N}\mathbf{Y} \\ \vdots & \ddots & \vdots \\ X_{N1}\mathbf{Y} & \cdots & X_{NN}\mathbf{Y} \end{pmatrix}. \tag{19}$$

For $\boldsymbol{x} \in \mathbb{R}^N$, the outer product is

$$\boldsymbol{x}\boldsymbol{x}^\intercal = \begin{pmatrix} x_1^2 & x_1x_2 & \cdots & x_1x_N \\ x_2x_1 & x_2^2 & \cdots & x_2x_N \\ \vdots & \vdots & \ddots & \vdots \\ x_Nx_1 & x_Nx_2 & \cdots & x_N^2 \end{pmatrix}. \tag{20}$$

For notational brevity, when no confusion arises, we occasionally suppress full parameter dependence. For example, $\Delta_t \equiv 1 - e^{-2t}$ is sometimes written simply as Δ.

The range of indices in summation and product symbols is omitted when its extremes are clear from the context, i.e. they were specified in previous equations.

B Further related works

Dynamical transitions in diffusion models. The recent success of generative diffusion models has promoted a growing effort to interpret their behavior through the lens of statistical mechanics, particularly in the high-dimensional limit where mean-field techniques become informative. Early works in this direction have analyzed how diffusion models reconstruct the target data distribution through a sequence of dynamical transitions that coincide with ergodicity-breaking events in an effective diffusion potential [2, 4, 6]. [7] characterizes the dynamical phases of diffusion models—namely speciation and collapse—while [3] highlighted speciation timescale separations for centered groups whose difference stems from higher-order moments. Still within the study of sampling dynamics in diffusion models, [1, 40, 47, 55, 60] investigated the evolution of the score function's geometry in space and time. In particular, [1, 60] demonstrated that: directions of higher variance in the data distribution are sampled earlier, as the score function progressively reconstructs the latent data structure; at later stages, data points that are strongly aligned with these high-variance directions are preferentially memorized, reflecting a temporal progression that parallels the earlier emergence of generalization.

Memorization and generalization in score learning. Another line of research has applied similar tools to study how diffusion models generalize the target distribution and memorize data points when parametrized by simple neural network architectures. In particular, [8, 16] study how a RF

model [44] fits the score function of a diffusion model when the data distribution is a unimodal multivariate Gaussian. Memorization is connected to the local density of training points in data space in [38], with low-density regions shown to be more likely to be memorized. The transition from memorization to generalization as dataset size increases has been widely documented [27], with early stopping acting as a crucial implicit regularisation mechanism driven by time-scale separation in overparameterized networks [8, 15, 58]: in particular, [58] identifies the mechanism behind the time-scale separation, analysing the spectrum of the Gram matrix of the Neural Tangent Kernel. Notably, the universal time-scale separation between feature learning and overfitting demonstrated in two-layer regression networks [39] closely parallels these memorization dynamics and transitions, with the crucial difference that overfitting in diffusion models is always malign [32, 57]. Memorization has also been explicitly detected by analysing the Hessian of the score in models with outliers [25], conceptually echoing earlier findings on spurious correlations [49]. Unlike these studies, which primarily focus on unimodal distributions or isolated microscopic outliers, our work provides a solvable analytical framework for macroscopic data mixtures, explicitly deriving how the combination of class imbalance and cluster variance dictates distinct, class-specific critical times for both learning and memorization.

Furthermore, multiple studies address score learning when the model is parametrized by a linear or shallow denoiser. [36] showed that while PCA fails to recover signals in a spiked cumulant model, shallow non-linear autoencoders succeed. Other works [37, 62] focus on the dynamical manner in which linear denoisers learn the data covariance. [62] shows that directions of large variance are learned first during training, consistent with the fact that such directions are sampled earlier in the diffusion dynamics [60]. Previous studies characterized linear auto-encoders as correlation machines utilizing power iteration for generative modeling [64]. Within more complex architectures, U-Net diffusion models yield multiplicative compositional representations [33], whereas exact ODEs defined the online SGD dynamics of diffusion models utilizing two-layer autoencoders [10, 11]. Although these frameworks identify variance as a driver of learning speed, they omit the role of heterogeneous datasets and their contribution to learning. We extend this understanding by demonstrating that while variance is the dominant force, other contributing factors, such as centroid norm and class imbalance, play a crucial role, becoming the decisive factors that dictates the learning sequence when class variances are comparable.

Data imbalance and algorithmic bias. Despite recent theoretical advances [24, 42, 49, 50], analytical literature on the effect of heterogeneous datasets and data imbalance remains extremely sparse. Empirically, it has been shown that generative models can exacerbate biases relating to demographic traits such as race, gender, and age, even when trained on balanced datasets [41]. While the evolution of bias across modern diffusion models is highly dataset-dependent [61], mechanistic interpretability approaches suggest that these biases often stem from neurons encoding for a multitude of semantic concepts—termed polysemantic—within the architectures [31, 51]. However, a rigorous theoretical understanding of *why* and *when* these models prioritize majority over minority classes during training is lacking. Our work bridges this critical gap, providing exact analytical predictions demonstrating that data structure and severe class imbalance force the features of undersampled modes to emerge at distinct, lower inference times.

Appendices for theory

C The solution of the dynamics

We consider the gradient flow (GF) limit of the gradient descent of the readout weights:

$$\mathbf{A}_{k+1} = \mathbf{A}_k - \eta \frac{\partial \mathcal{E}_{train}}{\partial \mathbf{A}}|_{\mathbf{A}=\mathbf{A}_k}. \tag{21}$$

where the training loss is defined in (3). We consider the small learning rate limit: we set $\eta \approx \widetilde{\eta}\, d\tau$, so that $\mathbf{A}_{k+1} - \mathbf{A}_k \approx d\mathbf{A}$ and

$$d\mathbf{A} = -\widetilde{\eta} \frac{\partial \mathcal{E}_{train}}{\partial \mathbf{A}}\bigg|_{\mathbf{A}=\mathbf{A}_\tau} d\tau.$$

The GF equation then reads

$$\dot{\mathbf{A}} = -\widetilde{\eta} \frac{\partial \mathcal{E}_{train}}{\partial \mathbf{A}}\bigg|_{\mathbf{A}=\mathbf{A}(\tau)} \equiv -2\widetilde{\eta} \frac{\Delta_t}{N} \left[\mathbf{A}\mathbf{U} + \frac{1}{\sqrt{\Delta_t}} \mathbf{V}^\intercal \right]. \tag{22}$$

with $\mathbf{U}$ and $\mathbf{V}$ defined in (8), (9). The GF equation is an ODE for the matrix $\mathbf{A}$, and thus admits an explicit solution. Let us consider the Singular Value Decomposition (SVD) of $\mathbf{A} = \mathbf{M}_L^\intercal \mathbf{S}_A \mathbf{M}_R$, with $\mathbf{M}_L \in \mathbb{R}^{N\times N}$ and $\mathbf{M}_R \in \mathbb{R}^{P\times P}$ two orthogonal matrices: then the GF equation becomes

$$\dot{\mathbf{S}}_A = -2\widetilde{\eta} \frac{\Delta_t}{N} \left[\mathbf{S}_A \widehat{\mathbf{U}} + \frac{1}{\sqrt{\Delta_t}} \widehat{\mathbf{V}}^\intercal \right], \tag{23}$$

with $\widehat{\mathbf{U}} = \mathbf{M}_R \mathbf{U} \mathbf{M}_R^\intercal$ and $\widehat{\mathbf{V}} = \mathbf{M}_R \mathbf{V} \mathbf{M}_L^\intercal$. Since the matrix $\mathbf{S}_A$ is non-zero only on the diagonalwith indices $i = 1, \dots, N$, we have the set of scalar equations for the singular values $\sigma_A^{(i)}$

$$\dot{\sigma}_A^{(i)} = -2\widetilde{\eta} \frac{\Delta_t}{N} \left[\sigma_A^{(i)} \widehat{\lambda}_i + \frac{1}{\sqrt{\Delta_t}} \widehat{v}_i \right] \tag{24}$$

where $\widehat{\lambda}_i \equiv \widehat{U}_{ii}$ and $\widehat{v}_i = \widehat{V}_{ii}$. They admit the solutions

$$\sigma_A^{(i)} = -\frac{1}{\sqrt{\Delta_t}} \frac{\widehat{v}_i}{\widehat{\lambda}_i} (1 - e^{-2\widetilde{\eta}\frac{\Delta_t}{N}\widehat{\lambda}_i \tau}) + \sigma_A^{(i)}(0) e^{-2\widetilde{\eta}\frac{\Delta_t}{N}\widehat{\lambda}_i \tau} \qquad i = 1, \dots, N \tag{25}$$

Taking the matrix form of this last equation and projecting back, we obtain

$$\mathbf{A}(\tau) = -\frac{1}{\sqrt{\Delta_t}} \mathbf{V}^\intercal \mathbf{U}^{-1} \left[\mathbf{I}_P - e^{-\frac{2\widetilde{\eta}\Delta_t}{N}\mathbf{U}\tau} \right] + \mathbf{A}(0) e^{-\frac{2\widetilde{\eta}\Delta_t}{N}\mathbf{U}\tau}. \tag{26}$$

By setting $\mathbf{A}(0) = \mathbf{0}$ and $\widetilde{\eta} = N/\Delta_t$, we obtain Eq. (7).

D The GEP for non-centered data

Let us consider a vector field $\boldsymbol{\varphi}$ as follows:

$$\boldsymbol{\varphi} = \phi\left(\frac{1}{\sqrt{N}} \mathbf{W}\boldsymbol{x} \right) \tag{27}$$

where ϕ is a non-linear function applied entrywise to the vector $\boldsymbol{u} = \frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{x}$, $\boldsymbol{x} \in \mathbb{R}^N$ is a variable with mean $\boldsymbol{m}$ and covariance $\boldsymbol{\Sigma}$, while $\mathbf{W} \in \mathbb{R}^{P\times N}$ is a random matrix with normal entries. By the central limit theorem, for $N \to \infty$ the vector $\boldsymbol{u}$ is a Gaussian vector with the following mean and covariance matrix:

$$\boldsymbol{\mu}_u = \frac{1}{\sqrt{N}} \mathbf{W}\boldsymbol{m} \tag{28}$$

$$\boldsymbol{\Sigma}_u = \frac{1}{N} \mathbf{W}\boldsymbol{\Sigma}\mathbf{W}^\intercal \tag{29}$$

We now evaluate the first two cumulants of $\boldsymbol{\varphi}$: the first reads

$$\boldsymbol{\alpha} \equiv \mathbb{E}_{\boldsymbol{x}\sim\mathcal{P}_x}\left[\boldsymbol{\varphi}\right] = \mathbb{E}_{\boldsymbol{u}\sim\mathcal{N}(\boldsymbol{\mu}_u,\boldsymbol{\Sigma}_u)}\left[\phi(\boldsymbol{u})\right] = \mathbb{E}_z\left[\phi(\sigma z\mathbf{1}_P + \boldsymbol{\mu}_u)\right] + \mathcal{O}_N(N^{-1/2}) \tag{30}$$

where we indicated with $\mathcal{P}_x$ the distribution of $\boldsymbol{x}$ and

$$\sigma^2 = \frac{1}{N}\,\mathrm{Tr}\,\boldsymbol{\Sigma} \tag{31}$$

and $\mathbf{1}_P \equiv (1,\dots,1)^\intercal \in \mathbb{R}^P$. We used the fact that for $N \gg 1$ one has $\Sigma^{(u)}_{pp} \simeq \frac{1}{N}\,\mathrm{Tr}\,\boldsymbol{\Sigma} = \sigma^2$. For the non-diagonal entries of the second cumulant, we use the Mehler kernel formula and obtain

$$\mathbb{E}_{\boldsymbol{x}\sim\mathcal{P}_x}\left[\varphi_p\varphi_q\right] - \mathbb{E}_{\boldsymbol{x}\sim\mathcal{P}_x}\left[\varphi_p\right]\mathbb{E}_{\boldsymbol{x}\sim\mathcal{P}_x}\left[\varphi_q\right] =$$

$$\sum_{n=0}^{\infty}\frac{1}{n!}\left(\frac{\Sigma^{(u)}_{pq}}{\sigma^2}\right)^n \mathbb{E}_z\left[\mathrm{He}_n(z)\phi(z\sigma+\mu_u^{(p)})\right]\mathbb{E}_z\left[\mathrm{He}_n(z)\phi(z\sigma+\mu_u^{(q)})\right] - \alpha_p\alpha_q \tag{32}$$

$$= \frac{1}{\sigma^2}\gamma_p\gamma_q\Sigma^{(u)}_{pq} + \mathcal{O}_N(N^{-1}), \qquad \boldsymbol{\gamma} = \mathbb{E}_z\left[z\,\phi(z\sigma\mathbf{1}_P + \boldsymbol{\mu}_u)\right]$$

Note that the off-diagonal elements of the covariance vanish in the high-dimensional limit $\Sigma^{(u)}_{pq} = \mathcal{O}(1/\sqrt{N})$. The second moment of the diagonal entries reads

$$\boldsymbol{\beta} \equiv \mathbb{E}_{\boldsymbol{x}\sim\mathcal{P}_x}\left[\boldsymbol{\varphi}^{\odot 2}\right] = \mathbb{E}_{\boldsymbol{u}\sim\mathcal{N}(\boldsymbol{\mu}_u,\boldsymbol{\Sigma}_u)}\left[\phi^{\odot 2}(\boldsymbol{u})\right] = \mathbb{E}_z\left[\phi^2(\sigma z\mathbf{1}_P + \boldsymbol{\mu}_u)\right] + \mathcal{O}_N(N^{-1/2}). \tag{33}$$

The second order cumulant is then

$$\mathbb{E}_{\boldsymbol{x}\sim\mathcal{P}_x}\left[\boldsymbol{\varphi}\boldsymbol{\varphi}^\intercal\right] - \mathbb{E}_{\boldsymbol{x}\sim\mathcal{P}_x}\left[\boldsymbol{\varphi}\right]\mathbb{E}_{\boldsymbol{x}\sim\mathcal{P}_x}\left[\boldsymbol{\varphi}\right]^\intercal \simeq \frac{1}{\sigma^2}\boldsymbol{\gamma}\boldsymbol{\gamma}^\intercal \odot \boldsymbol{\Sigma}_u + \mathrm{diag}(\boldsymbol{\beta} - \boldsymbol{\gamma}^{\odot 2} - \boldsymbol{\alpha}^{\odot 2}). \tag{34}$$

The first two cumulants of the non-linear field $\boldsymbol{\varphi}$ are equivalent to those of a linear field $\boldsymbol{\varphi}_{gep}$ with Gaussian noise:

$$\boxed{\boldsymbol{\varphi}_{gep}(\boldsymbol{u}) = \boldsymbol{\alpha} + \frac{1}{\sigma}(\boldsymbol{u}-\boldsymbol{\mu}_u)\odot\boldsymbol{\gamma} + \boldsymbol{\eta}\,\odot\,(\boldsymbol{\beta} - \boldsymbol{\gamma}^{\odot 2} - \boldsymbol{\alpha}^{\odot 2})^{\odot\frac{1}{2}} \qquad \boldsymbol{\eta}\sim\mathcal{N}(\mathbf{0},\mathbf{I}_P)} \tag{35}$$

Note that if we set $\boldsymbol{m} = \mathbf{0}$ and take an odd activation ϕ, then $\boldsymbol{\alpha} = \mathbf{0}$, $\boldsymbol{\mu}_u = \mathbf{0}$ and we obtain the known formula for the GEP for centered variables:

$$\boldsymbol{\varphi}_{gep}(\boldsymbol{u}) = \frac{1}{\sigma}\gamma\,\boldsymbol{u} + \sqrt{\beta-\gamma^2}\,\boldsymbol{\eta} \tag{36}$$

We can apply the previous GEP formula to a generic multimodal distribution that admits a pure-density decomposition [3]. Indeed, one has

$$\mathbb{E}_{\boldsymbol{x}\sim\sum_{c=1}^C b_c\mathcal{P}_c(\cdot)}\left[\boldsymbol{\varphi}\boldsymbol{\varphi}^\intercal\right] = \sum_{c=1}^C b_c\,\mathbb{E}_{\boldsymbol{x}\sim\mathcal{N}(\mathbf{m}_c,\boldsymbol{\Sigma}_c),\,\boldsymbol{\eta}_c\sim\mathcal{N}(\mathbf{0},\mathbf{I}_P)}\left[\boldsymbol{\varphi}^{(c)}_{gep}\boldsymbol{\varphi}^{(c)\,\intercal}_{gep}\right], \tag{37}$$

where we apply the GEP to each component of the mixture:

$$\boldsymbol{\varphi}^{(c)}_{gep}(\boldsymbol{u}) = \boldsymbol{\alpha}_c + \frac{1}{\sigma_c}(\boldsymbol{u}-\boldsymbol{\mu}_u^{(c)})\odot\boldsymbol{\gamma}_c + \boldsymbol{\eta}_c\,\odot\,(\boldsymbol{\beta}_c - \boldsymbol{\gamma}_c^{\odot 2} - \boldsymbol{\alpha}_c^{\odot 2})^{\odot\frac{1}{2}} \qquad \boldsymbol{\eta}_c\sim\mathcal{N}(\mathbf{0},\mathbf{I}_P). \tag{38}$$

E The GEP of feature correlation matrices

We derive the GEP expressions for the feature-feature and feature-noise correlation matrices in Eqs. (8), (9), that we rewrite here for reader's convenience:

$$\mathbf{U} = \sum_{c=1}^C b_c\mathbf{U}_c, \qquad \mathbf{U}_c = \frac{1}{M_c}\sum_{\nu\in I_c}\mathbb{E}_{\boldsymbol{\xi}}\left[\phi\left(\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{x}_\nu(t)\right)\phi\left(\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{x}_\nu(t)\right)^\intercal\right],$$

$$\mathbf{V} = \sum_{c=1}^C b_c\mathbf{V}_c, \qquad \mathbf{V}_c = \frac{1}{M_c}\sum_{\nu\in I_c}\mathbb{E}_{\boldsymbol{\xi}}\left[\phi\left(\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{x}_\nu(t)\right)\boldsymbol{\xi}^\intercal\right],$$

$$\boldsymbol{x}_\nu(t) = \boldsymbol{x}_\nu e^{-t} + \sqrt{\Delta_t}\boldsymbol{\xi}.$$

We also derive the GEP of the corresponding population matrices,

$$\widetilde{\mathbf{U}} = \sum_{c=1}^{C} b_c \widetilde{\mathbf{U}}_c, \qquad \widetilde{\mathbf{U}}_c = \mathbb{E}_{\boldsymbol{\xi},\boldsymbol{x}(0)\sim\mathcal{N}_c}\left[\phi\left(\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{x}(t)\right)\phi\left(\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{x}(t)\right)^{\intercal}\right] \tag{39}$$

$$\widetilde{\mathbf{V}} = \sum_{c=1}^{C} b_c \widetilde{\mathbf{V}}_c, \qquad \widetilde{\mathbf{V}}_c = \mathbb{E}_{\boldsymbol{\xi},\boldsymbol{x}(0)\sim\mathcal{N}_c}\left[\phi\left(\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{x}(t)\right)\boldsymbol{\xi}^{\intercal}\right] \tag{40}$$

that will be used in Appendix I. Our derivation follows the steps of those in [8, 17], focusing on mixtures of isotropic modes, i.e., with $\Sigma_c \equiv \sigma_c^2 \mathbf{I}_N$.

E.1 The GEP of U

We begin by deriving the diagonal elements of $\mathbf{U_c}$: they can be obtained straightforwardly from the concentration of the empirical mean in (8)

$$U_{\mathrm{c}}^{(qq)} = \frac{1}{M_c}\sum_{\nu\in I_c}\mathbb{E}_{\boldsymbol{\xi}}\left[\phi^2\left(\frac{1}{\sqrt{N}}\sum_{j=1}^{N} W_{qj}x_j^{(\nu)}(t)\right)\right] = \widetilde{\beta}_c^{(q)}(t) + \mathcal{O}_M(M^{-1/2}) \tag{41}$$

$$\widetilde{\boldsymbol{\beta}}_c(t) = \mathbb{E}_z\left[\phi^{\odot 2}\left(z\,\Gamma_c(t)\mathbf{1}_P + e^{-t}\boldsymbol{\mu}_c\right)\right].$$

We introduced the variance of class c at time t:

$$\Gamma_c^2(t) = \sigma_c^2 e^{-2t} + \Delta_t. \tag{42}$$

As for the off-diagonal elements of $\mathbf{U}_c$, one first introduces the diffusion noise latent vector

$$\mathrm{u} = \frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{\xi} \tag{43}$$

which has zero mean and a covariance with unit diagonal entries and off-diagonal elements $\Sigma_u^{(pq)} = \frac{1}{N}\sum_{i=1}^{N} W_{pi}W_{qi}$. Then, by applying the Mehler kernel formula to (8), we get

$$U_c^{(pq)} = \frac{1}{M_c}\sum_{\nu\in I_c}\phi_0^{(t)}\left(\frac{1}{\sqrt{N}}\sum_{i=1}^{N} W_{pi}x_i^{(\nu)}\right)\phi_0^{(t)}\left(\frac{1}{\sqrt{N}}\sum_{i=1}^{N} W_{qi}x_i^{(\nu)}\right)$$

$$+\frac{\Sigma_u^{(pq)}}{M_c}\sum_{\nu\in I_c}\phi_1^{(t)}\left(\frac{1}{\sqrt{N}}\sum_{i=1}^{N} W_{pi}x_i^{(\nu)}\right)\phi_1^{(t)}\left(\frac{1}{\sqrt{N}}\sum_{i=1}^{N} W_{qi}x_i^{(\nu)}\right) + \mathcal{O}_N(N^{-1}) \tag{44}$$

with

$$\phi_0^{(t)}(x) = \mathbb{E}_z\left[\phi\left(e^{-t}x + \sqrt{\Delta_t}z\right)\right] \qquad \phi_1^{(t)}(x) = \mathbb{E}_z\left[z\,\phi\left(e^{-t}x + \sqrt{\Delta_t}z\right)\right]$$

The truncation of the expansion was possible because $\Sigma_u^{(pq)} = \frac{1}{N}\sum_i W_{pi}W_{qi} = \mathcal{O}(N^{-1/2})$. Let us first evaluate the second term in (44): we use the following Hermite expansion of ϕ for $c_0 > 0$ and $c_1 \in \mathbb{R}$

$$\phi(c_0 x + c_1) = \sum_{n=0}^{\infty}\frac{h_n}{n!}\,\mathrm{He}_n(x), \qquad h_n = \mathbb{E}_z\left[\mathrm{He}_n(z)\,\phi(c_0 z + c_1)\right]. \tag{45}$$

With all this, we get

$$\frac{1}{M_c}\sum_{\nu\in I_c}\phi_1^{(t)}\left(\frac{1}{\sqrt{N}}\sum_{i=1}^{N}W_{pi}x_i^{(\nu)}\right)\phi_1^{(t)}\left(\frac{1}{\sqrt{N}}\sum_{i=1}^{N}W_{pi}x_i^{(\nu)}\right)^{\intercal}$$

$$\simeq \mathbb{E}_{u\sim\mathcal{N}(\mu_c^{(p)},\sigma_c^2),\,y}\left[y\,\phi\left(e^{-t}u_p+\sqrt{\Delta_t}y\right)\right]\mathbb{E}_{u\sim\mathcal{N}(\mu_c^{(q)},\sigma_c^2),\,y}\left[y\,\phi\left(e^{-t}u_q+\sqrt{\Delta_t}y\right)\right]$$

then

$$\mathbb{E}_{u\sim\mathcal{N}(\mu_c^{(p)},\sigma_c^2),\,y}\left[y\,\phi\left(e^{-t}u_p+\sqrt{\Delta(t)}y\right)\right]=\sum_{n=0}^{\infty}\frac{h_n^{(p)}}{n!}\mathbb{E}_{y,\,\widetilde{u}}\left[y\,\mathrm{He}_n\left(\frac{e^{-t}\sigma_c\widetilde{u}+\sqrt{\Delta_t}y}{\Gamma_t^c}\right)\right] \tag{46}$$

$$=\frac{\sqrt{\Delta(t)}}{\Gamma_c(t)}\widetilde{\gamma}_c^{(p)}(t),$$

$$\widetilde{\boldsymbol{\gamma}}_c(t)=\mathbb{E}_z\left[z\,\phi\left(z\,\Gamma_c(t)\mathbf{1}_P+e^{-t}\boldsymbol{\mu}_c\right)\right],$$

and thus (44) becomes

$$U_c^{(pq)}=\frac{1}{M_c}\sum_{\nu\in I_c}\phi_0^{(t)}\left(\frac{1}{\sqrt{N}}\sum_{i=1}^{N}W_{pi}x_i^{(\nu)}\right)\phi_0^{(t)}\left(\frac{1}{\sqrt{N}}\sum_{i=1}^{N}W_{qi}x_i^{(\nu)}\right)+\frac{\Delta_t}{\Gamma_c^2(t)}\widetilde{\gamma}_c^{(p)}\widetilde{\gamma}_c^{(q)}\Sigma_u^{(pq)}.$$

To evaluate the first term in (44), we use the GEP of Eq. (38):

$$\phi_0\left(\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{x}^{(\nu)}\right)\overset{\text{GEP}}{\to}\boldsymbol{G}_c^{(\nu)} \tag{47}$$

$$\boldsymbol{G}_c^{(\nu)}\equiv\boldsymbol{\alpha}_c+\frac{1}{\sigma_c\sqrt{N}}\mathbf{W}(\boldsymbol{x}^{(\nu)}-\boldsymbol{m}_c)\odot\boldsymbol{\gamma}_c+(\boldsymbol{\beta}_c-\boldsymbol{\alpha}_c^{\odot 2}-\boldsymbol{\gamma}_c^{\odot 2})^{\odot\frac{1}{2}}\odot\boldsymbol{\eta}_\nu^c,$$

where the vectors of Hermite coefficients can be computed using their definition in Appendix D and the definition of ϕ_0 in (44):

$$\boldsymbol{\alpha}_c(t)=\mathbb{E}_z\left[\phi_0(z\,\sigma_c\mathbf{1}_P+\boldsymbol{m}_c)\right]=\mathbb{E}_z\left[\phi(z\,\Gamma_c(t)\,\mathbf{1}_P+e^{-t}\boldsymbol{\mu}_c)\right] \tag{48}$$

$$\boldsymbol{\beta}_c(t)=\mathbb{E}_z\left[\phi_0^{\odot 2}(z\,\sigma_c\,\mathbf{1}+\boldsymbol{m}_c)\right]$$

$$=\mathbb{E}_{u,v\sim\mathcal{N}(0,\boldsymbol{\Sigma}_c^{(2)}(t))}\left[\phi(\Gamma_c(t)u\,\mathbf{1}_P+e^{-t}\boldsymbol{\mu}_c)\odot\phi(\Gamma_c(t)v\,\mathbf{1}_P+e^{-t}\boldsymbol{\mu}_c)\right]$$

with

$$\boldsymbol{\Sigma}_c^{(2)}(t)=\begin{pmatrix}1 & \frac{e^{-2t}\sigma_c^2}{(\Gamma_c(t))^2}\\ \frac{e^{-2t}\sigma_c^2}{(\Gamma_c(t))^2} & 1\end{pmatrix} \tag{49}$$

$$\boldsymbol{\gamma}_c(t)=\mathbb{E}_z\left[z\,\phi_0(z\,\sigma_c\mathbf{1}_P+\boldsymbol{m}_c)\right]=\frac{\sigma_c e^{-t}}{\Gamma_c(t)}\widetilde{\boldsymbol{\gamma}}_c(t) \tag{50}$$

Putting everything together, the GEP form of the matrix $\mathbf{U}$ is equal to

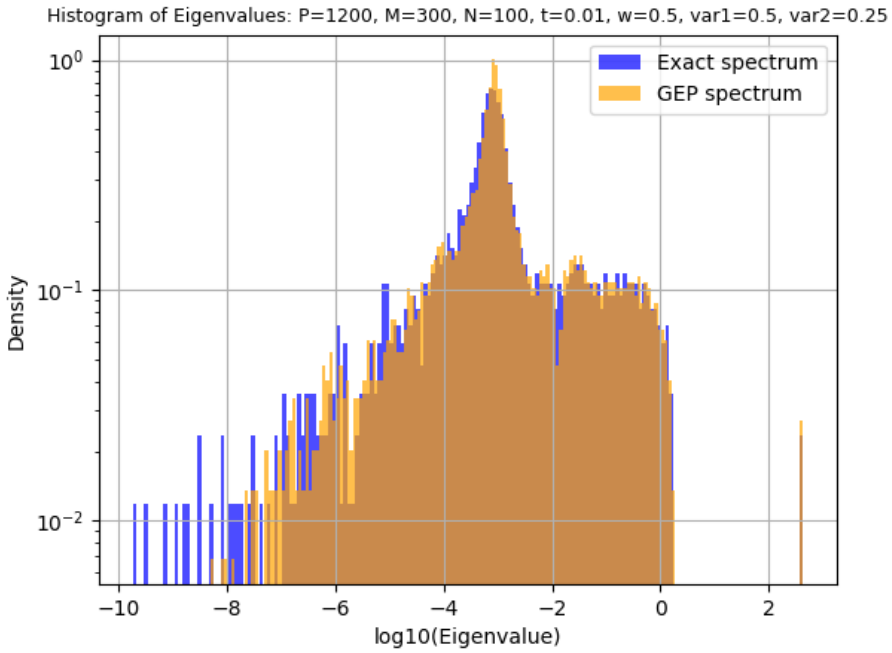


Figure 7: Comparison of the GEP derived in Eq. (51) with the spectrum of U obtained through a Monte Carlo estimation of (8).

$$
\begin{aligned}
\mathbf{U} &\sim \sum_{c=1}^{C} b_c \mathbf{U}_c^{(gep)}, \\
\mathbf{U}_c^{(gep)} &= \frac{1}{M_c} \sum_{\nu \in I_c} \boldsymbol{G}_c^{(\nu)}(t) \boldsymbol{G}_c^{(\nu)}(t)^\intercal + \frac{\Delta(t)}{N \sigma_c^2 e^{-2t}} \boldsymbol{\gamma}_c(t) \boldsymbol{\gamma}_c(t)^\intercal \odot \mathbf{W}\mathbf{W}^\intercal \\
&\quad + \operatorname{diag}\left(\widetilde{\boldsymbol{\beta}}_c(t) - \boldsymbol{\beta}_c(t) - \frac{\Delta_t}{\sigma_c^2 e^{-2t}} \boldsymbol{\gamma}_c(t)^{\odot 2} \right), \\
\boldsymbol{G}_c^{(\nu)} &= \boldsymbol{\alpha}_c + \frac{1}{\sigma_c \sqrt{N}} \mathbf{W}(\boldsymbol{x}^{(\nu)} - \boldsymbol{m}_c) \odot \boldsymbol{\gamma}_c + (\boldsymbol{\beta}_c - \boldsymbol{\alpha}_c^{\odot 2} - \boldsymbol{\gamma}_c^{\odot 2})^{\odot \frac{1}{2}} \odot \boldsymbol{\eta}_\nu^c
\end{aligned}
\tag{51}
$$

In (10) we set

$$
\widehat{\mathbf{W}}_c \equiv \boldsymbol{\gamma}_c \mathbf{1}_N^\intercal \odot \mathbf{W}, \qquad \boldsymbol{d}_c = \widetilde{\boldsymbol{\beta}}_c(t) - \boldsymbol{\beta}_c(t) - \frac{\Delta_t}{\sigma_c^2 e^{-2t}} \boldsymbol{\gamma}_c(t)^{\odot 2}. \tag{52}
$$

Note that the GEP matrix can be decomposed in terms of a finite-norm matrix $\widehat{\mathbf{U}}_c^{(gep)}$ plus a low rank extensive perturbation

$$
\mathbf{U}_c^{(gep)} = \widehat{\mathbf{U}}_c^{(gep)} + \boldsymbol{\alpha}_c \boldsymbol{\alpha}_c^\intercal + \left[\frac{\boldsymbol{\alpha}_c}{M_c} \sum_{\nu \in I_c} (\mathbf{G}_c^{(\nu)} - \boldsymbol{\alpha}_c)^\intercal + \frac{1}{M_c} \sum_{\nu \in I_c} (\mathbf{G}_c^{(\nu)} - \boldsymbol{\alpha}_c) \boldsymbol{\alpha}_c^\intercal \right] \tag{53}
$$

The perturbation is extensive because its first term is a rank-one matrix of norm $\mathcal{O}_P(P)$, while the other is $\mathcal{O}_P(1)$. The first term in particular is related to the projection of the centroids in the space of features: this can be seen explicitly in the large t limit:

$$
\boldsymbol{\alpha}_c \approx \gamma \, e^{-t} \boldsymbol{\mu}_c. \tag{54}
$$

where in the last equation $\gamma = \mathbb{E}_z\left[z\phi(z)\right]$. We tested the goodness of the GEP in (51): in Fig. 7, we compare the spectrum of $\mathbf{U}_{\rm gep}$ with that of the exact $\mathbf{U}$, which was estimated with Monte-Carlo.

E.2 The GEP of $\widetilde{\mathrm{U}}$

To compute the GEP of the corresponding population matrix, we simply use the GEP on the vector field

$$\boldsymbol{\varphi} = \phi\left(\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{x}(t)\right)$$

with $\boldsymbol{x}(t)$ sampled from $\mathcal{N}(\cdot, \boldsymbol{m}_c e^{-t}, \Gamma_c^2(t)\mathbf{I}_N)$. We get

$$\boldsymbol{\varphi}_{gep}^{(c)} = \boldsymbol{\alpha}_c + \frac{1}{\sigma_c\sqrt{N}}\mathbf{W}(\boldsymbol{x}^{(\nu)} - \boldsymbol{m}_c)\odot\boldsymbol{\gamma}_c + (\widetilde{\boldsymbol{\beta}}_c - \boldsymbol{\alpha}_c^{\odot 2} - \widetilde{\boldsymbol{\gamma}}_c^{\odot 2})^{\odot\frac{1}{2}}\odot\boldsymbol{\eta}_\nu^c \qquad \boldsymbol{\eta}_\nu^c \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_P) \tag{55}$$

and thus the GEP of $\widetilde{\mathbf{U}}$ is obtained by averaging the $\boldsymbol{\varphi}_{gep}^{(c)}\boldsymbol{\varphi}_{gep}^{(c)\intercal}$ over the $\boldsymbol{x} \sim \mathcal{N}_c$ and the $\boldsymbol{\eta}$, obtaining

$$\boxed{\begin{aligned}\widetilde{\mathbf{U}} &\sim \sum_{c=1}^{C} b_c \widetilde{\mathbf{U}}_c^{(gep)},\\ \widetilde{\mathbf{U}}_c^{(gep)} &= \boldsymbol{\alpha}_c\boldsymbol{\alpha}_c^\intercal + \frac{1}{N}\mathbf{W}\mathbf{W}^\intercal\odot\widetilde{\boldsymbol{\gamma}}_c\widetilde{\boldsymbol{\gamma}}_c^\intercal + \mathrm{diag}(\widetilde{\boldsymbol{\beta}}_c - \boldsymbol{\alpha}_c^{\odot 2} - \widetilde{\boldsymbol{\gamma}}_c^{\odot 2}).\end{aligned}} \tag{56}$$

One can easily verify that by averaging $\mathbf{U}_c^{(gep)}$ over the dataset and the GEP noise, one directly obtains (56).

It is insightful to relate $\mathbf{U}$ to its population version $\widetilde{\mathbf{U}}$: with a few straightforward manipulations, one can show that

$$\mathbf{U}_c^{gep} = \widetilde{\mathbf{U}}_c^{gep} + \delta\mathbf{U}_c^{(gep)}$$

$$\begin{aligned}\delta\mathbf{U}_c^{(gep)} &= \frac{1}{M_c}\sum_{\nu\in I_c}\boldsymbol{G}_c^{(\nu)}(t)\boldsymbol{G}_c^{(\nu)}(t)^\intercal - \frac{1}{N}\mathbf{W}\mathbf{W}^\intercal\odot\boldsymbol{\gamma}_c\boldsymbol{\gamma}_c^\intercal \\ &+ \mathrm{diag}\left(\widetilde{\boldsymbol{\beta}}_c - \boldsymbol{\beta}_c - \frac{\Delta_t}{\sigma_c^2 e^{-2t}}\boldsymbol{\gamma}_c^{\odot 2}\right) - \mathrm{diag}(\widetilde{\boldsymbol{\beta}}_c - \boldsymbol{\alpha}_c^{\odot 2} - \widetilde{\boldsymbol{\gamma}}_c^{\odot 2}) - \boldsymbol{\alpha}_c\boldsymbol{\alpha}_c^\intercal\end{aligned} \tag{57}$$

The generalization during training is due to the population matrix $\widetilde{\mathbf{U}} \sim \sum_c b_c\widetilde{\mathbf{U}}_c^{gep}$, while memorization originates from the matrix $\delta\mathbf{U} \sim \sum_c b_c\delta\mathbf{U}_c^{gep}$. Proceeding similarly to [8], one can verify that $\|\widetilde{\mathbf{U}} - \sum_c b_c\boldsymbol{\alpha}_c\boldsymbol{\alpha}_c^\intercal\|_F = \mathcal{O}(\chi_p)$ and $\|\delta\mathbf{U}\|_F = \mathcal{O}(\chi_p/\sqrt{\chi_m})$, where $\|\cdot\|_F$ is the Frobenius norm.

E.3 The GEP of $\mathbf{V}$

We take formula (9) and rewrite the latent vector $\boldsymbol{u}_\nu(t) = \frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{x}_\nu(t)$, where $\nu \in I_c$, as

$$\boldsymbol{u}_\nu(t) = \boldsymbol{u}_\nu + e^{-t}\boldsymbol{\mu}_c \tag{58}$$

The vector $\boldsymbol{u}_\nu$ is a Gaussian vector with zero mean and covariance $\boldsymbol{\Sigma}_{uu} = \frac{\Gamma_c^2(t)}{N}\mathbf{W}\mathbf{W}^\intercal$. The cross covariance with the noise reads $\boldsymbol{\Sigma}_{u\xi} = \frac{\sqrt{\Delta_t}}{\sqrt{N}}\mathbf{W}$. Then the Mehler kernel formula applied to (9) yields

$$\begin{aligned}V_c^{(pj)} &= \frac{1}{M_c}\sum_{\nu\in I_c}\sum_{n=0}^{\infty}\frac{(\Sigma_{u\xi}^{(pj)})^n}{n!}\mathbb{E}_z\left[\phi\left(z\Gamma_c(t) + e^{-t}\mu_c^{(p)}\right)\mathrm{He}_n(z)\right]\mathbb{E}_z\left[z\,\mathrm{He}_n(z)\right] \\ &= \sqrt{\frac{\Delta_t}{N}}\frac{\widetilde{\gamma}_c^{(p)}}{\Gamma_c(t)}W_{pj} + \mathcal{O}_N(N^{-1})\end{aligned} \tag{59}$$

Then the GEP for $\mathbf{V}$ reads

$$\boxed{\begin{aligned}\mathbf{V} &\sim \sum_{c=1}^{C} b_c \mathbf{V}_c^{(gep)} \\ \mathbf{V}_c^{(gep)} &= \sqrt{\frac{\Delta_t}{N}} \frac{\widetilde{\boldsymbol{\gamma}}_c \mathbf{1}_N^\intercal}{\Gamma_c(t)} \odot \mathbf{W}.\end{aligned}} \tag{60}$$

E.4 The GEP of $\widetilde{\mathbf{V}}$

It is easy to see from (59) that the GEP of $\widetilde{\mathbf{V}}$ is the same as that of $\mathbf{V}$, up to finite-size corrections: so we write

$$\boxed{\begin{aligned}\widetilde{\mathbf{V}} &\sim \sum_{c=1}^{C} b_c \widetilde{\mathbf{V}}_c^{(gep)} \\ \widetilde{\mathbf{V}}_c^{(gep)} &= \mathbf{V}_c^{(gep)}.\end{aligned}} \tag{61}$$

F The matrix U at large t

We derive Eq. (12) through a large t expansion—or equivalently, a small e^{-t} expansion. We start with the Hermite coefficients: for any component $p = 1, \dots, P$

$$\begin{aligned}\alpha_c^{(p)} &= \mathbb{E}_z\left[\phi(\Gamma_c z + \mu_c^{(p)} e^{-t})\right] = \int \frac{dz}{(2\pi\Gamma_c^2)^{1/2}} e^{-\frac{(z-\mu_c^{(p)} e^{-t})^2}{2\Gamma_c^2}} \phi(z) \\ &= \mathbb{E}_z\left[\phi(z)\right] + \mu_c^{(p)} \mathbb{E}_z\left[z\phi(z)\right] e^{-t} + \mathcal{O}(e^{-2t})\end{aligned} \tag{62}$$

$$\begin{aligned}\widetilde{\beta}_c^{(p)} &= \mathbb{E}_z\left[\phi^2(\Gamma_c z + \mu_c^{(p)} e^{-t})\right] = \int \frac{dz}{(2\pi\Gamma_c^2)^{1/2}} e^{-\frac{(z-\mu_c^{(p)} e^{-t})^2}{2\Gamma_c^2}} \phi^2(z) \\ &= \mathbb{E}_z\left[\phi^2(z)\right] + \mu_c^{(p)} \mathbb{E}_z\left[z\phi^2(z)\right] e^{-t} + \mathcal{O}(e^{-2t})\end{aligned} \tag{63}$$

$$\begin{aligned}\beta_c^{(p)} &= \mathbb{E}_z\left[\phi^2(\Gamma_c z + \mu_c^{(p)} e^{-t})\right] \\ &= \int \frac{dz_1 dz_2}{2\pi\Gamma_c^2\sqrt{\det \boldsymbol{\Sigma}_c^{(2)}}} e^{-\frac{(z_1-\mu_c^{(p)} e^{-t})^2}{2\Gamma_c^2 \det \boldsymbol{\Sigma}_c^{(2)}} - \frac{(z_2-\mu_c^{(p)} e^{-t})^2}{2\Gamma_c^2 \det \boldsymbol{\Sigma}_c^{(2)}} + \frac{(z_1-\mu_c^{(p)} e^{-t})(z_2-\mu_c^{(p)} e^{-t})\sigma_c^2 e^{-2t}}{\Gamma_c^4 \det \boldsymbol{\Sigma}_c^{(2)}}} \phi(z_1)\phi(z_2) \\ &= \mathbb{E}_z\left[\phi(z)\right]^2 + 2\mu_c^{(p)} \mathbb{E}_z\left[\phi(z)\right] \mathbb{E}_z\left[z\phi(z)\right] e^{-t} + \mathcal{O}(e^{-2t})\end{aligned} \tag{64}$$

$$\begin{aligned}\widetilde{\gamma}_c^{(p)} &= \mathbb{E}_z\left[z\phi(\Gamma_c z + \mu_c^{(p)} e^{-t})\right] = \int \frac{dz}{(2\pi\Gamma_c^2)^{1/2}} e^{-\frac{(z-\mu_c^{(p)} e^{-t})^2}{2\Gamma_c^2}} z\phi(z) \\ &= \mathbb{E}_z\left[z\phi(z)\right] + \mu_c^{(p)} \mathbb{E}_z\left[z^2\phi(z)\right] e^{-t} + \mathcal{O}(e^{-2t})\end{aligned} \tag{65}$$

$$\gamma_c^{(p)} = \frac{\sigma_c e^{-t}}{\Gamma_c} \widetilde{\gamma}_c^{(p)} = \sigma_c e^{-t} \mathbb{E}_z\left[z\phi(z)\right] + \mathcal{O}(e^{-2t}) \tag{66}$$

where we also expanded $\Gamma_c(t) = \sqrt{(\sigma_c^2 - 1)e^{-2t} + 1} \approx 1 + \mathcal{O}(e^{-2t})$. Let us define $\alpha = \mathbb{E}_z\left[\phi(z)\right]$, $\beta = \mathbb{E}_z\left[\phi^2(z)\right]$, $\gamma = \mathbb{E}_z\left[z\phi(z)\right]$. At leading order in e^{-t}, the GEP features read

$$\boldsymbol{G}_{c,\nu} = \alpha \mathbf{1}_P + \gamma e^{-t}\boldsymbol{\mu}_c + \frac{e^{-t}\gamma}{\sqrt{N}}\mathbf{W}(\boldsymbol{x}_\nu - \boldsymbol{m}_c) + \mathcal{O}(e^{-2t}) \tag{67}$$

The other terms in Eq. (51) expand as follows:

$$\frac{\Delta(t)}{N\sigma_c^2 e^{-2t}}\boldsymbol{\gamma}_c(t)\boldsymbol{\gamma}_c(t)^\intercal \odot \mathbf{W}\mathbf{W}^\intercal = \frac{\gamma^2}{N}\mathbf{W}\mathbf{W}^\intercal + \mathcal{O}(e^{-2t}) \tag{68}$$

$$\operatorname{diag}\left(\widetilde{\boldsymbol{\beta}}_c - \boldsymbol{\beta}_c - \frac{\Delta}{\sigma_c^2 e^{-2t}}\boldsymbol{\gamma}_c^{\odot 2}\right) = (\beta - \alpha^2 - \gamma^2)\mathbf{I}_P + \mathcal{O}(e^{-2t}) \tag{69}$$

We now assume that $t = \mathcal{O}(\log N)$, so that $e^{-t} = \mathcal{O}(N^{-1})$: then, we can neglect all corrections but those in the finite-rank perturbation in Eq. (51), and the matrix $\mathbf{U}$ becomes

$$\mathbf{U} = (\alpha \mathbf{1}_P + \gamma e^{-t}\boldsymbol{\mu}_c)(\alpha \mathbf{1}_P + \gamma e^{-t}\boldsymbol{\mu}_c)^\intercal + \frac{\gamma^2}{N}\mathbf{W}\mathbf{W}^\intercal + (\beta - \alpha^2 - \gamma^2)\mathbf{I}_P \tag{70}$$

For $\phi = \tanh$, $\alpha = 0$ and we obtain Eq. (12).

G The role of the activation function

In this Appendix we explain how the choice of the activation function affects the learning dynamics in the DSM of Eq. (3): first, we consider the simple case of a linear activation; then, we characterize how the generalization and memorization gaps of sections 3.2.1 change when using a ReLU activation instead of $\tanh$.

Linear activation The role of the activation function is encoded in the Hermite coefficients $\{\boldsymbol{\alpha}_c, \boldsymbol{\beta}_c, \widetilde{\boldsymbol{\beta}}_c, \boldsymbol{\gamma}_c\}_{c=1}^C$, defined in eqs (48),(49),(41), (50). For a linear activation, the Hermite coefficients simplify to

$$\boldsymbol{\alpha}_c = \mathbb{E}_z\left[\phi(\Gamma_c z\mathbf{1} + \boldsymbol{\mu}_c e^{-t})\right] \to \boldsymbol{\mu}_c e^{-t} \tag{71}$$

$$\boldsymbol{\beta}_c = \mathbb{E}_{u,v\sim\mathcal{N}(\mathbf{0},\boldsymbol{\Sigma}_c^{(2)}(t))}\left[\phi(\Gamma_c u\mathbf{1} + \boldsymbol{\mu}_c e^{-t}) \odot \phi(\Gamma_c v\mathbf{1} + \boldsymbol{\mu}_c e^{-t})\right] \to \boldsymbol{\mu}_c^{\odot 2}e^{-2t} + \sigma_c^2 e^{-2t}\mathbf{1} \tag{72}$$

$$\widetilde{\boldsymbol{\beta}}_c = \mathbb{E}_z\left[\phi^2(\Gamma_c z\mathbf{1} + \boldsymbol{\mu}_c e^{-t})\right] \to \boldsymbol{\mu}_c^{\odot 2}e^{-2t} + \Gamma_c^2\mathbf{1} \tag{73}$$

$$\widetilde{\boldsymbol{\gamma}}_c = \mathbb{E}_z\left[z\phi(\Gamma_c z\mathbf{1} + \boldsymbol{\mu}_c e^{-t})\right] \to \Gamma_c\mathbf{1} \Longrightarrow \boldsymbol{\gamma}_c \to \sigma_c e^{-t}\mathbf{1} \tag{74}$$

and the GEP correlation matrices $\mathbf{U}_{\mathrm{gep}}$ and $\widetilde{\mathbf{U}}_c^{(\mathrm{gep})}$, $\mathbf{V}_c$, $c = 1, \dots, C$ simply reduce respectively to the empirical and population data correlation matrices at diffusion time t:

$$\mathbf{U}_{\mathrm{gep}} \to \sum_c b_c \boldsymbol{\mu}_c\boldsymbol{\mu}_c^\intercal e^{-2t} + \frac{1}{MN}\sum_{\nu=1}^M \mathbf{W}\left[e^{-2t}(\boldsymbol{x}_\nu - \boldsymbol{m}_c)(\boldsymbol{x}_\nu - \boldsymbol{m}_c)^\intercal + \Delta\mathbf{I}_N\right]\mathbf{W}^\intercal \tag{75}$$

$$\mathbf{V}_c^{(\mathrm{gep})} \to \sqrt{\frac{\Delta}{N}}\mathbf{W} \tag{76}$$

$$\widetilde{\mathbf{U}}_c^{(\mathrm{gep})} \to \boldsymbol{\mu}_c\boldsymbol{\mu}_c^\intercal e^{-2t} + \frac{\Gamma_c(t)}{N}\mathbf{W}\mathbf{W}^\intercal \tag{77}$$

For linear activations, the bulks of $\mathbf{U}$ and $\widetilde{\mathbf{U}}_c$ are not affected by either the magnitude or the alignment of the centroids, nor by class sampling ratios b_c: these parameters only affect the finite-rank perturbation term $\sum_c b_c\boldsymbol{\mu}_c\boldsymbol{\mu}_c^\intercal$ of the projected centroids $\boldsymbol{\mu}_c \equiv \frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{m}_c$. According to the learning curves formula of Eq. (11), only the bulks of the matrices $\mathbf{U}, \widetilde{\mathbf{U}}_c$, $c = 1, \dots, C$ bear finite contributions to these curves, as any finite-rank perturbation yields sub-leading $\mathcal{O}(1/N)$ corrections: we conclude that generalization and memorization gaps induced by differences in centroid geometry and sampling imbalance are a non-linear effect in the activation function.

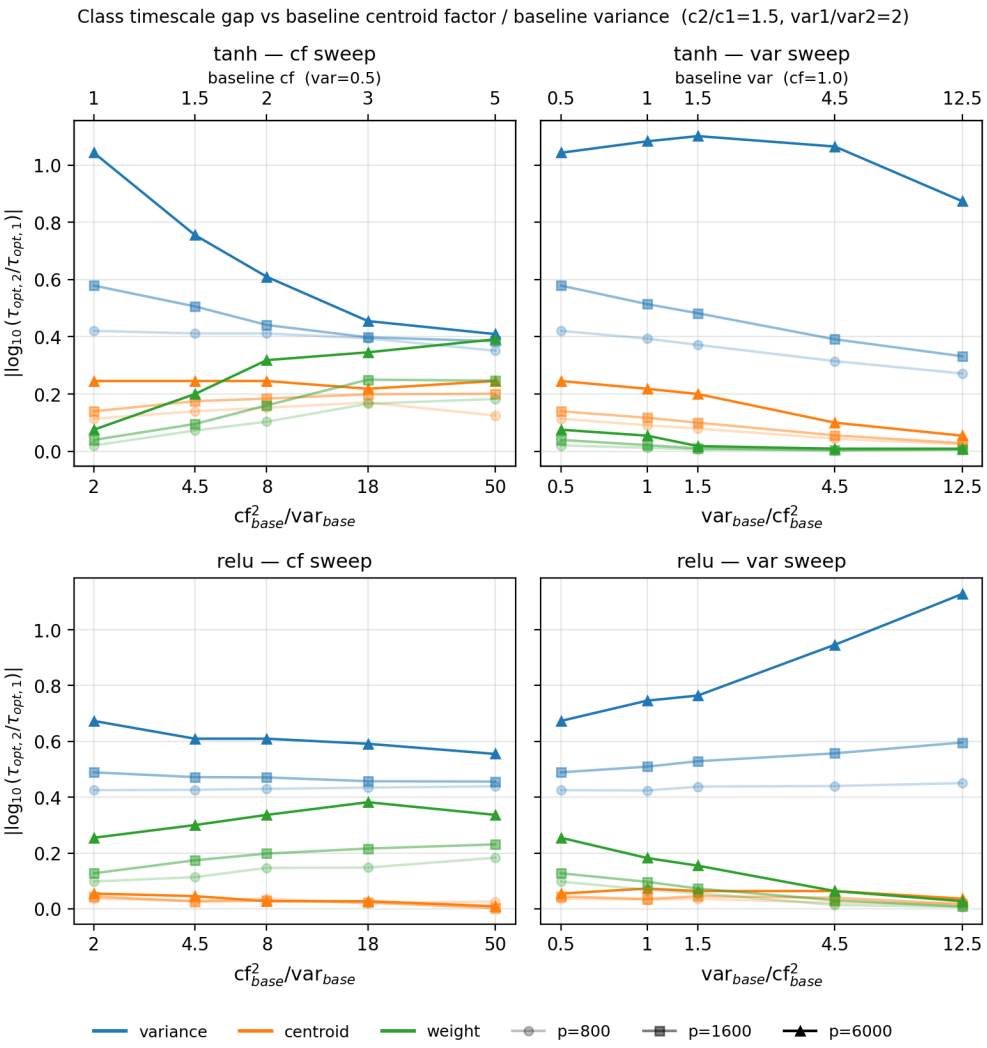


Figure 8: Generalization log-gaps versus baseline ratios: top is for a $\tanh$ activation, bottom for a ReLU activation; left panels refer to a sweep in baseline centroid factors $\mathrm{cf}_b = 1.0, 1.5, 2.0, 3.0, 5.0$, right panels to a sweep in baseline variance $\sigma_b^2 = 0.5, 1.0, 1.5, 4.5, 12.5$. Curves are obtained from the semi-analytical curves of Eqs.(148). Parameters are $N = 100$, $M = 3000$, $P = 800, 1600, 6000$.

Generalization and memorization gaps In this paragraph, we compute the generalization and memorization gaps of section 3.2.1 for an RF model with a ReLU activation function and compare them with the $\tanh$ activation used in the rest of the work. We consider the three ablations of Fig. 3 with ratios between variances and centroid factors $\sigma_1^2/\sigma_2^2 = 2$, $\mathrm{cf}_2/\mathrm{cf}_1$ and evaluate the gaps first for different baseline centroids $\mathrm{cf}_b \equiv \mathrm{cf}_1 = 1.0, 1.5, 2.0, 3.0, 5.0$ and then for different baseline variances $\sigma_b^2 \equiv \sigma_1^2 = 0.5, 1.0, 1.5, 4.5, 12.5$. In Figs. 8,9 we show respectively the generalization and memorization log-gaps, $\log\left(\tau_g^{(2)}/\tau_g^{(1)}\right)$ and $\log\left(\tau_m^{(2)}/\tau_m^{(1)}\right)$, versus $\mathrm{cf}_b^2/\sigma_b^2$ (left panels) and σ_b^2/cf^2 (right panels): for each of the ablations of Fig. 3, we see how the gaps are affected by varying the baseline centroid factor cf_b (left panels) and the baseline variance σ_b^2 (right panels). For $\mathrm{cf}_b \lessapprox \sigma_b$, both plots confirm result 2: variance is the dominating factor for the learning gaps, with centroid norm and sampling rates playing a secondary role. Only for $\mathrm{cf}_b \gg \sigma_b$ does the role of centroid norm become comparable to that of variance. This limit does not correspond to realistic datasets statistics.

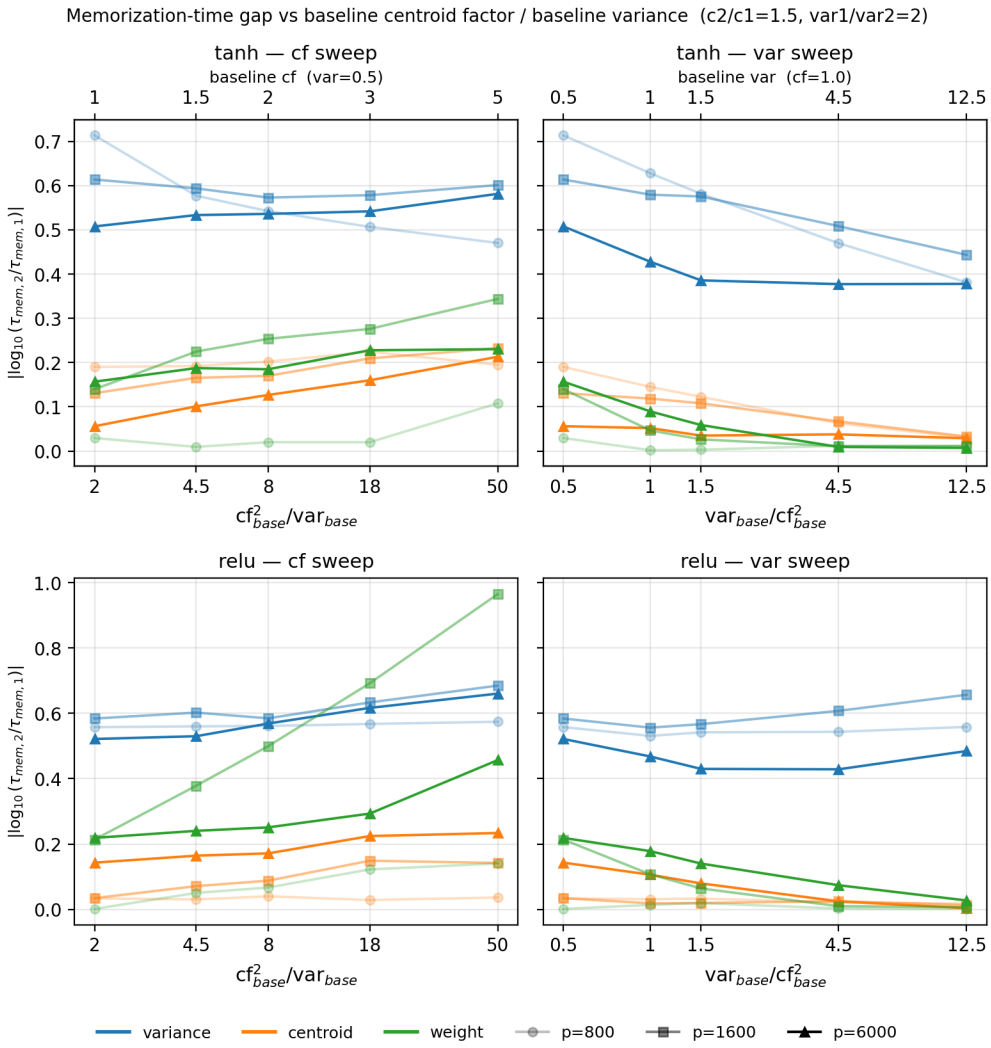


Figure 9: Memorization log-gaps versus baseline ratios: top is for a $\tanh$ activation, bottom for a ReLU activation; left panels refer to a sweep in baseline centroid factors $\mathrm{cf}_b = 1.0, 1.5, 2.0, 3.0, 5.0$, right panels to a sweep in baseline variance $\sigma_b^2 = 0.5, 1.0, 1.5, 4.5, 12.5$. Curves are obtained from the semi-analytical curves of Eqs.(148). Parameters are $N = 100$, $M = 3000$, $P = 800, 1600, 6000$.

H Spectral equations of U

The spectrum of the matrix $\mathbf{U}$ is fully characterized by the resolvent matrix

$$\mathcal{G}_U(z) = (\mathbf{U} - z\mathbf{I}_P)^{-1} \equiv \sum_{q=1}^{P} \frac{\boldsymbol{\psi}_q \boldsymbol{\psi}_q^\intercal}{\lambda_q - z}, \qquad z \in \mathbb{C} \setminus \mathrm{Spec}(\mathbf{U}) \tag{78}$$

where $\mathrm{Spec}(\cdot)$ is the set of eigenvalues of a matrix. Since the matrix $\mathbf{U}$ is random through the dataset $\mathbf{X}$ and the random features matrix $\mathbf{W}$, the resolvent is a random matrix itself. However, its first matrix moment concentrates in the high-dimensional limit:

$$\lim_{P\to\infty} \left| \frac{1}{P} \operatorname{Tr} \mathcal{G}_U(z) - \mathbb{E}_{\mathbf{X},\mathbf{W}} \left[\frac{1}{P} \operatorname{Tr} \mathcal{G}_U(z) \right] \right| = 0 \tag{79}$$

$$g_U(z) = \frac{1}{P} \operatorname{Tr} \mathcal{G}_U(z) \overset{P\to\infty}{\Longrightarrow} g(z) = \int \frac{d\lambda\, \rho(\lambda)}{\lambda - z}$$

We shall call the asymptotic limit of the resolvent first moment the *resolvent function*: it is the Stieltjes transform [43] of the density of eigenvalues $\rho(\lambda)$. From the statistics of the resolvent matrix, we can evaluate the statistical properties of the eigenvalues and the eigenvectors of $\mathbf{U}$: in particular, from the resolvent function in (79) we can compute the density of eigenvalues through the Sokhotski-Plemelj formula, also known as the inverse Stieltjes transform:

$$\rho(\lambda) = \frac{1}{\pi} \lim_{\epsilon \to 0_+} \operatorname{Im} g(\lambda + i\epsilon) \tag{80}$$

To evaluate the resolvent function, we use computational techniques of statistical physics. We consider the Edwards-Jones formula [14]:

$$g(z) = \lim_{P\to\infty} \frac{1}{P} \mathbb{E}_{\mathbf{X},\mathbf{W}} \left[\operatorname{Tr} (\mathbf{U} - z\mathbf{I}_P)^{-1} \right] = -\frac{\partial}{\partial z} \lim_{P\to\infty} \frac{1}{P} \mathbb{E}_{\mathbf{X},\mathbf{W}} \left[\operatorname{Tr} \log (\mathbf{U} - z\mathbf{I}_P) \right]$$

$$= -\frac{\partial}{\partial z} \lim_{P\to\infty} \frac{1}{P} \mathbb{E}_{\mathbf{X},\mathbf{W}} \left[\log \det (\mathbf{U} - z\mathbf{I}_P) \right] = 2\frac{\partial}{\partial z} \lim_{P\to\infty} \frac{1}{P} \mathbb{E}_{\mathbf{X},\mathbf{W}} \left[\log \mathcal{Z}_U \right], \tag{81}$$

$$\mathcal{Z}_U = \int d\boldsymbol{\psi}\, e^{-\frac{1}{2}\boldsymbol{\psi}^\intercal (\mathbf{U} - z\mathbf{I}_P)\boldsymbol{\psi}}.$$

H.1 Replica computation of the resolvent function

The goal of a statistical physics computation is to represent high-dimensional systems in terms of few, low-dimensional parameters: in the case of random matrix calculations, these are resolvent functions of random matrices. Depending on the calculation, the final result can be characterized by several different resolvents. In particular, the main one and objective of this calculation is the resolvent function (79), which it is easy to see from (81) that it is given by

$$g(z) = \lim_{P\to\infty} \frac{1}{P} \mathbb{E}_{\mathbf{X},\mathbf{W}} \left[\int d\boldsymbol{\psi}\, \boldsymbol{\psi}^\intercal \boldsymbol{\psi}\, e^{-\frac{1}{2}\boldsymbol{\psi}^\intercal (\mathbf{U} - z\mathbf{I}_P)\boldsymbol{\psi}} \right]. \tag{82}$$

To evaluate the average over the log in (81), we use the Replica trick [9]:

$$\log \mathcal{Z}_U = \lim_{n\to 0} \frac{\mathcal{Z}_U^n - 1}{n} \Longrightarrow \mathbb{E}_{\mathbf{X},\mathbf{W}} \left[\log \mathcal{Z}_U \right] = \lim_{n\to 0} \frac{\mathbb{E}_{\mathbf{X},\mathbf{W}} \left[\mathcal{Z}_U^n \right] - 1}{n}$$

with

$$\begin{aligned} \mathbb{E}_{\mathbf{X},\mathbf{W}} \left[\mathcal{Z}_U^n \right] &= \mathbb{E}_{\mathbf{X},\mathbf{W}} \left[\left(\int d\boldsymbol{\psi}\, e^{-\frac{1}{2}\boldsymbol{\psi}^\intercal (\mathbf{U} - z\mathbf{I}_P)\boldsymbol{\psi}} \right)^n \right] \\ &\equiv \mathbb{E}_{\mathbf{X},\mathbf{W}} \left[\prod_{a=1}^n \int d\boldsymbol{\psi}_a\, e^{-\frac{1}{2}\boldsymbol{\psi}_a^\intercal (\mathbf{U} - z\mathbf{I}_P)\boldsymbol{\psi}_a} \right]. \end{aligned} \tag{83}$$

Then, we replace $\mathbf{U}$ with its GEP in (51):

$$\mathbb{E}_{\mathbf{X},\mathbf{W}} \left[\prod_{a=1}^n \int d\boldsymbol{\psi}_a\, e^{-\frac{1}{2}\boldsymbol{\psi}_a^\intercal (\mathbf{U} - z\mathbf{I}_P)\boldsymbol{\psi}_a} \right] = \mathbb{E}_{\{\mathbf{X}_c, \mathbf{H}_c\}_{c=1}^C, \mathbf{W}} \left[\prod_{a=1}^n \int d\boldsymbol{\psi}_a\, e^{-\frac{1}{2}\boldsymbol{\psi}_a^\intercal (\sum_{c=1}^C b_c \mathbf{U}_c^{(gep)} - z\mathbf{I}_P)\boldsymbol{\psi}_a} \right]$$

where we indicate with $\mathbf{H}_c \in \mathbb{R}^{P\times M_c}$ the matrix having as columns the noise vectors of the GEP of the different classes. Finally, we consider Eq. (53): since we are mostly interested in the absolutely continuous part of the spectrum of $\mathbf{U}^{(gep)}$, we perform the computation for the matrix $\widehat{\mathbf{U}}^{(gep)} = \sum_{c=1}^C b_c \widehat{\mathbf{U}}_c^{(gep)}$, given that this matrix does not have a low-rank perturbation. Indeed, the two matrices have the same absolutely continuous spectrum, differing only by the presence of outliers in $\mathbf{U}_{gep}$. Thus, in what follows, we proceed with the computation of the following expectation:

$$\overline{\mathcal{Z}^n} = \mathbb{E}_{\{\mathbf{X}_c, \mathbf{H}_c\}_{c=1}^C, \mathbf{W}} \left[\prod_{a=1}^n \int d\boldsymbol{\psi}_a\, e^{-\frac{1}{2}\boldsymbol{\psi}_a^\intercal (\sum_{c=1}^C b_c \widehat{\mathbf{U}}_c^{(gep)} - z\mathbf{I}_P)\boldsymbol{\psi}_a} \right]. \tag{84}$$

We set $\widetilde{\boldsymbol{G}}_c^{(\nu)} = \boldsymbol{G}_c^{(\nu)} - \boldsymbol{\alpha}_c$ (compare with Eq. (51)) and expand:

$$\overline{\mathcal{Z}^n} = \mathbb{E}_{\{\mathbf{X}_c,\mathbf{H}_c\}_{c=1}^C,\mathbf{W}}\Bigg[\int d\boldsymbol{\Psi}\,\prod_{a=1}^n e^{-\frac{1}{2}\sum_{c=1}^C b_c\boldsymbol{\psi}_a^\intercal(\operatorname{diag}\boldsymbol{\varsigma}_c - z\mathbf{I}_P)\boldsymbol{\psi}_a - \frac{1}{2M}\boldsymbol{\psi}_a^\intercal\sum_{c=1}^C\widetilde{\mathbf{G}}_c\widetilde{\mathbf{G}}_c^\intercal\boldsymbol{\psi}_a}$$

$$\times e^{-\frac{\Delta}{2N}\boldsymbol{\psi}_a^\intercal\left(\sum_{c=1}^C\frac{b_c}{\Gamma_c^2}\widetilde{\boldsymbol{\gamma}}_c\widetilde{\boldsymbol{\gamma}}_c^\intercal\right)\odot\mathbf{W}\mathbf{W}^\intercal\boldsymbol{\psi}_a}\Bigg]$$

where we set $d\boldsymbol{\psi}_1\cdots d\boldsymbol{\psi}_n = d\boldsymbol{\Psi}$ and defined the vectors

$$\boldsymbol{\varsigma}_c = \widetilde{\boldsymbol{\beta}}_c - \boldsymbol{\beta}_c - \frac{\Delta_t}{\sigma_c^2 e^{-2t}}\boldsymbol{\gamma}_c^{\odot 2}. \tag{85}$$

We do the disorder averages over the data, the GEP noise, and the random features step by step: to this end, it is convenient to compact the terms in the GEP features:

$$\widetilde{G}_c^{p\nu} = \frac{\gamma_c^{(p)}}{\sigma_c\sqrt{N}}\sum_{j=1}^N W_{pj}(x_j^{(c,\nu)} - m_j^{(c)}) + \widetilde{\eta}_{p\nu}^{(c)}\,, \qquad \widetilde{\boldsymbol{\eta}}_\nu^{(c)} \sim \mathcal{N}(\cdot\,;\mathbf{0},\operatorname{diag}(\boldsymbol{h}_c^{\odot 2})) \tag{86}$$

where we set $\boldsymbol{h}_c^{\odot 2} = \boldsymbol{\beta}_c - \boldsymbol{\alpha}_c^{\odot 2} - \boldsymbol{\gamma}_c^{\odot 2}$. We then group terms in the GEP covariances in terms of their disorder dependence:

$$\frac{1}{M}\big(\widetilde{\boldsymbol{G}}_c\widetilde{\boldsymbol{G}}_c^\intercal\big)_{pq} = \Bigg\{\frac{\gamma_c^{(p)}\,\gamma_c^{(q)}}{M\,\sigma_c^2\,N}\sum_{i=1}^N\sum_{j=1}^N\sum_{\nu=1}^{M_c} W_{pi}\,W_{qj}\;\big(x_i^{(c,\nu)} - m_i^{(c)}\big)\big(x_j^{(c,\nu)} - m_j^{(c)}\big)$$

$$+\left[\frac{\gamma_c^{(p)}}{M\sqrt{N}\,\sigma_c}\sum_{i=1}^N W_{qi}\sum_{\nu=1}^{M_c}\big(x_i^{(c,\nu)} - m_i^{(c)}\big)\widetilde{\eta}_{q\nu}^{(c)} + \frac{\gamma_c^{(q)}}{M\sqrt{N}\,\sigma_c}\sum_{i=1}^N W_{pi}\sum_{\nu=1}^{M_c}\big(x_i^{(c,\nu)} - m_i^{(c)}\big)\widetilde{\eta}_{p\nu}^{(c)}\right]\Bigg\}$$

$$+\left\{\frac{1}{M}\sum_{v=1}^{M_c}\widetilde{\eta}_{p\nu}^{(c)}\,\widetilde{\eta}_{q\nu}^{(c)}\right\} \equiv A_{pq}^{(X_c,\widetilde{H}_c,W)} + A_{pq}^{(\widetilde{H}_c,W)}.$$

Note that each of the vector GEP coefficients $\boldsymbol{\alpha}_c, \boldsymbol{\beta}_c, \boldsymbol{\gamma}_c, \widetilde{\boldsymbol{\beta}}_c$ depends on the random features $\mathbf{W}$ through the projected centroids $\boldsymbol{\mu}_c \equiv (1/\sqrt{N})\mathbf{W}\mathbf{m}_c$; see their definitions in Appendix E. We now proceed with the computations of the disorder averages.

Average over the data. The average over the data $\{\boldsymbol{X}_c\}_{c=1}^C$ can be factorized across the different classes' data. The generic c integral is a standard multi-variate integral

$$\mathbb{E}_{\mathbf{X}_c}\left[e^{-\frac{1}{2}\sum_a\boldsymbol{\psi}_a^\intercal\mathbf{A}^{(X_c,\widetilde{H}_c,W)}\boldsymbol{\psi}_a}\right] =$$

$$\prod_{\nu=1}^{M_c}\int\frac{d\boldsymbol{x}_\nu^{(c)}}{(2\pi\sigma_c^2)^{N/2}}\,e^{-\frac{1}{2\sigma_c^2}\left(\boldsymbol{x}_\nu^{(c)}-\mathbf{m}_c\right)^\intercal\left(\mathbf{I}_N+\frac{1}{NM}\mathbf{W}^\intercal\sum_a(\boldsymbol{\gamma}_c\odot\boldsymbol{\psi}_a)(\boldsymbol{\gamma}_c\odot\boldsymbol{\psi}_a)^\intercal\mathbf{W}\right)\left(\boldsymbol{x}_\nu^{(c)}-\boldsymbol{m}_c\right)}$$

$$\times e^{\frac{1}{\sqrt{N}\sigma_c M}\sum_a\boldsymbol{\psi}_a^\intercal\widetilde{\boldsymbol{\eta}}_\nu^{(c)}[(\boldsymbol{\psi}_a\odot\boldsymbol{\gamma}_c)^\intercal\mathbf{W}](\mathbf{x}_\nu^{(c)}-\mathbf{m}_c)}$$

$$= e^{\frac{1}{2NM^2}\sum_{\nu=1}^{M_c}\sum_{ab}(\boldsymbol{\psi}_a^\intercal\widetilde{\boldsymbol{\eta}}_\nu^{(c)})(\boldsymbol{\psi}_b^\intercal\widetilde{\boldsymbol{\eta}}_\nu^{(c)})(\mathbf{W}^\intercal\boldsymbol{\psi}_a\odot\boldsymbol{\gamma}_c)^\intercal(\mathbf{I}_N+\frac{1}{NM}\boldsymbol{J}_c)^{-1}(\mathbf{W}^\intercal\boldsymbol{\psi}_b\odot\boldsymbol{\gamma}_c) - \frac{M_c}{2}\log\det\left(\mathbf{I}_N+\frac{1}{NM}\boldsymbol{J}_c\right)},$$

$$\mathbf{J}_c \equiv \mathbf{W}^\intercal\sum_a(\boldsymbol{\gamma}_c\odot\boldsymbol{\psi}_a)(\boldsymbol{\gamma}_c\odot\boldsymbol{\psi}_a)^\intercal\mathbf{W}.$$

where we recall that $\chi_p \equiv P/N$. To make the $\mathbf{J}_c$ independent of the fields $\{\boldsymbol{\psi}_a\}$, we introduce the auxiliary fields $\boldsymbol{\omega}_{a;c} = \frac{1}{\sqrt{P}}\mathbf{W}^\intercal(\boldsymbol{\psi}_a \odot \boldsymbol{\gamma}_c))$ through delta functions

$$\begin{aligned} 1 &= \int \prod_{c=1}^{C} d\boldsymbol{\Omega}_c \delta(\sqrt{P}\boldsymbol{\omega}_{a;c} - \mathbf{W}^\intercal(\boldsymbol{\psi}_a \odot \boldsymbol{\gamma}_c)) \\ &= \int \prod_{c=1}^{C} d\boldsymbol{\Omega}_c \frac{d\widehat{\boldsymbol{\Omega}}_c}{(2\pi)^{nN}} e^{i\sqrt{P}\boldsymbol{\omega}_{a;c}^\intercal \widehat{\boldsymbol{\omega}}_{a;c} - i\widehat{\boldsymbol{\omega}}_{a;c}^\intercal \mathbf{W}^\intercal(\boldsymbol{\psi}_a \odot \boldsymbol{\gamma}_c)} \end{aligned} \tag{87}$$

where we set $d\boldsymbol{\Omega}_c = d\boldsymbol{\omega}_{1,c} \cdots d\boldsymbol{\omega}_{n,c}$ and $d\widehat{\boldsymbol{\Omega}}_c = d\widehat{\boldsymbol{\omega}}_{1,c} \cdots d\widehat{\boldsymbol{\omega}}_{n,c}$. After averaging over the data, Eq. (84) becomes

$$\begin{aligned} &\mathbb{E}_{\{\mathbf{X}_c, \mathbf{H}_c\}_{c=1}^C, \mathbf{W}} \Bigg[\int d\boldsymbol{\Psi} \prod_{a=1}^{n} e^{-\frac{1}{2}\sum_{c=1}^{C} b_c \boldsymbol{\psi}_a^\intercal (\operatorname{diag} \boldsymbol{\varsigma}_c - z\mathbf{I}_P)\boldsymbol{\psi}_a - \frac{1}{2M}\boldsymbol{\psi}_a^\intercal \sum_{c=1}^{C} \widetilde{\mathbf{G}}_c \widetilde{\mathbf{G}}_c^\intercal \boldsymbol{\psi}_a} \\ &\qquad \times e^{-\frac{\Delta}{2N}\boldsymbol{\psi}_a^\intercal \left(\sum_{c=1}^{C} \frac{b_c}{\Gamma_c^2} \widetilde{\boldsymbol{\gamma}}_c \widetilde{\boldsymbol{\gamma}}_c^\intercal \right) \odot \mathbf{W}\mathbf{W}^\intercal \boldsymbol{\psi}_a} \Bigg] \\ = \mathbb{E}_{\{\mathbf{H}_c\}_{c=1}^C, \mathbf{W}} \Bigg[&\int d\boldsymbol{\Psi} \int \prod_c d\boldsymbol{\Omega}_c \frac{d\widehat{\boldsymbol{\Omega}}_c}{(2\pi)^{Nn}} e^{i\sqrt{P}\sum_c \sum_a \boldsymbol{\omega}_{a;c}^\intercal \widehat{\boldsymbol{\omega}}_{a;c} - \frac{1}{2}\sum_c b_c \sum_a \boldsymbol{\psi}_a^\intercal (\operatorname{diag}\boldsymbol{\varsigma}_c - z\mathbf{I}_P)\boldsymbol{\psi}_a} \\ &\times e^{-\frac{P\Delta}{2Ne^{-2t}} \sum_c \sum_a \frac{b_c}{\sigma_c^2} \boldsymbol{\omega}_{a;c}^\intercal \boldsymbol{\omega}_{a;c} + \frac{\chi_p}{2M^2} \sum_c \sum_{\nu=1}^{M_c} \sum_{ab} (\boldsymbol{\psi}_a^\intercal \widetilde{\boldsymbol{\eta}}_\nu^{(c)})(\boldsymbol{\psi}_b^\intercal \widetilde{\boldsymbol{\eta}}_\nu^{(c)}) \boldsymbol{\omega}_{a;c}^\intercal (\mathbf{I}_N + \frac{\chi_p}{M}\boldsymbol{J}_c)^{-1} \boldsymbol{\omega}_{a;c}} \\ &\times e^{-\frac{M}{2}\sum_c b_c \log\det\left(\mathbf{I}_N + \frac{\chi_p}{M}\boldsymbol{J}_c\right) - i\sum_c \sum_a \widehat{\boldsymbol{\omega}}_{a;c}^\intercal \mathbf{W}^\intercal(\boldsymbol{\psi}_a \odot \boldsymbol{\gamma}_c)) - \frac{1}{2M}\sum_c \sum_a \boldsymbol{\psi}_a^\intercal \widetilde{\boldsymbol{\eta}}_\nu^{(c)} (\widetilde{\boldsymbol{\eta}}_\nu^{(c)})^\intercal \boldsymbol{\psi}_a} \Bigg]. \end{aligned}$$

Average over the GEP noise. Again, the average over the GEP noise $\{\widetilde{\mathbf{H}}_c\}_{c=1}^C$ can be factorized into the different classes

$$\mathbb{E}_{\widetilde{\mathbf{H}}_c} \left[e^{-\frac{1}{2}\sum_a \boldsymbol{\psi}_a^\intercal \mathbf{A}^{(\widetilde{H}_c, W)} \boldsymbol{\psi}_a + \frac{\chi_p}{2M^2} \sum_{\nu=1}^{M_c} \sum_{ab} \left(\widetilde{\boldsymbol{\eta}}_\nu^{(c)}\right)^\intercal \boldsymbol{\psi}_a \, \boldsymbol{\psi}_b^\intercal \widetilde{\boldsymbol{\eta}}_\nu^{(c)} \, \boldsymbol{\omega}_{a;c}^\intercal [\mathbf{I}_N + \frac{\chi_p}{M}\boldsymbol{J}_c]^{-1} \boldsymbol{\omega}_{b,c}} \right] =$$

$$\prod_{\nu=1}^{M_c} \int \frac{d\widetilde{\boldsymbol{\eta}}_\nu^{(c)} \, e^{-\frac{1}{2}(\widetilde{\boldsymbol{\eta}}_\nu^c)^\intercal \operatorname{diag}(\boldsymbol{h}_c^{\odot 2})^{-1} \widetilde{\boldsymbol{\eta}}_\nu^c}}{(2\pi)^{P/2}\sqrt{\det\left(\operatorname{diag}(\boldsymbol{h}_c^{\odot 2})\right)}} e^{-\frac{1}{2}\left(\widetilde{\boldsymbol{\eta}}_\nu^{(c)}\right)^\intercal \sum_{ab} \left[\frac{\delta_{ab}}{M} \boldsymbol{\psi}_a \boldsymbol{\psi}_a^\intercal - \frac{\chi_p}{2M^2} \boldsymbol{\psi}_a \, \boldsymbol{\omega}_{a;c}^\intercal [\mathbf{I}_N + \frac{\chi_p}{S}\boldsymbol{J}_c]^{-1} \boldsymbol{\omega}_{b,c} \boldsymbol{\psi}_b^\intercal \right] \widetilde{\boldsymbol{\eta}}_\nu^{(c)}} =$$

$$e^{-\frac{M_c}{2}\log\det\left(\mathbf{I}_P + \frac{1}{M}\operatorname{diag}(\boldsymbol{h}_c)\boldsymbol{\mathcal{J}}_c \operatorname{diag}(\boldsymbol{h}_c)\right)}$$

where

$$\boldsymbol{\mathcal{J}}_c(\boldsymbol{\psi}, \boldsymbol{\omega}_c) = \sum_{ab} \boldsymbol{\psi}_a \boldsymbol{\psi}_b^\intercal \left(\delta_{ab} - \frac{\chi_p}{M} \boldsymbol{\omega}_{a;c}^\intercal \left(\boldsymbol{I}_N + \frac{\chi_p}{M} \boldsymbol{J}_c'(\boldsymbol{\omega}_c) \right)^{-1} \boldsymbol{\omega}_{b,c} \right).$$

where we set $\boldsymbol{J}_c' = N\boldsymbol{J}_c$. We introduce the projected class centroids $\boldsymbol{\mu}_c = \frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{m}_c$ with a Dirac delta:

$$1 = \int \prod_{c=1}^{C} d\boldsymbol{\mu}_c \delta\left(\boldsymbol{\mu}_c - \frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{m}_c \right) = \int \prod_{c=1}^{C} d\boldsymbol{\mu}_c \frac{d\boldsymbol{\mu}_c}{(2\pi)^P} e^{i\widehat{\boldsymbol{\mu}}_c^\intercal \boldsymbol{\mu}_c - \frac{i}{\sqrt{N}}\widehat{\boldsymbol{\mu}}_c^\intercal \mathbf{W}\boldsymbol{m}_c} \tag{88}$$

After averaging over the data and the GEP noise, Eq. (84) becomes

$$\mathbb{E}_{\{\mathbf{X}_c,\mathbf{H}_c\}_{c=1}^C,\mathbf{W}}\Bigg[\int d\boldsymbol{\Psi}\prod_{a=1}^{n} e^{-\frac{1}{2}\sum_{c=1}^C b_c\boldsymbol{\psi}_a^\intercal(\operatorname{diag}\boldsymbol{\varsigma}_c - z\mathbf{I}_P)\boldsymbol{\psi}_a - \frac{1}{2M}\boldsymbol{\psi}_a^\intercal\sum_{c=1}^C\widetilde{\mathbf{G}}_c\widetilde{\mathbf{G}}_c^\intercal\boldsymbol{\psi}_a}$$

$$\times e^{-\frac{\Delta}{2N}\boldsymbol{\psi}_a^\intercal\left(\sum_{c=1}^C\frac{b_c}{\Gamma_c^2}\widetilde{\boldsymbol{\gamma}}_c\widetilde{\boldsymbol{\gamma}}_c^\intercal\right)\odot\mathbf{W}\mathbf{W}^\intercal\boldsymbol{\psi}_a}\Bigg]$$

$$= \mathbb{E}_{\mathbf{W}}\Bigg[\int d\boldsymbol{\Psi}\int\prod_c d\boldsymbol{\Omega}_c\frac{d\widehat{\boldsymbol{\Omega}}_c}{(2\pi)^{nN}}d\boldsymbol{\mu}_c\frac{d\boldsymbol{\mu}_c}{(2\pi)^P}e^{i\sqrt{P}\sum_c\sum_a\boldsymbol{\omega}_{a;c}^\intercal\widehat{\boldsymbol{\omega}}_{a;c}+i\sum_c\widehat{\boldsymbol{\mu}}_c^\intercal\boldsymbol{\mu}_c}$$

$$\times e^{-\frac{1}{2}\sum_c b_c\sum_a\boldsymbol{\psi}_a^\intercal(\operatorname{diag}\boldsymbol{\varsigma}_c - z\mathbf{I}_P)\boldsymbol{\psi}_a - \frac{P\Delta}{2Ne^{-2t}}\sum_c\sum_a\frac{b_c}{\sigma_c^2}\boldsymbol{\omega}_{a;c}^\intercal\boldsymbol{\omega}_{a;c} - \frac{M}{2}\sum_c b_c\log\det\left(\mathbf{I}_P+\frac{1}{M}\operatorname{diag}(\boldsymbol{h}_c)\boldsymbol{\mathcal{J}}_c\operatorname{diag}(\boldsymbol{h}_c)\right)}$$

$$\times e^{-\frac{M}{2}\sum_c b_c\log\det\left(\mathbf{I}_N+\frac{\chi_p}{M}\boldsymbol{J}_c\right) - i\sum_c\sum_a\widehat{\boldsymbol{\omega}}_{a;c}^\intercal\mathbf{W}^\intercal(\boldsymbol{\psi}_a\odot\boldsymbol{\gamma}_c) - \frac{i}{\sqrt{N}}\sum_c\widehat{\boldsymbol{\mu}}_c^\intercal\mathbf{W}\boldsymbol{m}_c}\Bigg].$$

Average over the random features. The average over the random features $\mathbf{W}$ is straightforward:

$$\mathbb{E}_W\left[e^{-i\sum_c\left[\frac{1}{\sqrt{N}}\widehat{\boldsymbol{\mu}}_c^\intercal\boldsymbol{W}\boldsymbol{m}_c+\sum_a(\boldsymbol{\gamma}_c\odot\boldsymbol{\psi}_a)^\intercal\boldsymbol{W}\widehat{\boldsymbol{\omega}}_{a;c}\right]}\right] =$$

$$\int\prod_{pj}\frac{dW_{pj}}{\sqrt{2\pi}}e^{-\frac{1}{2}\sum_{pj}W_{pj}^2 - i\sum_c\left[\frac{1}{\sqrt{N}}\widehat{\boldsymbol{\mu}}_c^\intercal\boldsymbol{W}\boldsymbol{m}_c+\sum_a(\boldsymbol{\gamma}_c\odot\boldsymbol{\psi}_a)^\intercal\boldsymbol{W}\widehat{\boldsymbol{\omega}}_{a;c}\right]} = \tag{89}$$

$$e^{-\frac{1}{2N}\sum_{c,c'}\boldsymbol{m}_c^\intercal\boldsymbol{m}_{c'}\widehat{\boldsymbol{\mu}}_c^\intercal\widehat{\boldsymbol{\mu}}_{c'} - \frac{1}{2}\sum_{c,c'}\sum_{ab}(\boldsymbol{\gamma}_c\odot\boldsymbol{\psi}_a)^\intercal(\boldsymbol{\gamma}_{c'}\odot\boldsymbol{\psi}_b)\,\widehat{\boldsymbol{\omega}}_{a;c}^\intercal\widehat{\boldsymbol{\omega}}_{b,c'} + o_P(P)}$$

The term that is $o_P(P)$ is $-\frac{1}{\sqrt{N}}\sum_{c,c'}\sum_a(\widehat{\boldsymbol{\omega}}_{a;c}^\intercal\boldsymbol{m}_c)\,(\boldsymbol{\gamma}_c\odot\boldsymbol{\psi}_a)^\intercal\widehat{\boldsymbol{\mu}}_{c'}$: the first dot yields an $O_P(\sqrt{P})$ contribution, the second an $\mathcal{O}_P(1)$ one, given that $\|\widehat{\boldsymbol{\omega}}_{a;c}\| = \mathcal{O}_P(1)$; then, the overall scaling of the last term is $\mathcal{O}_P(1)$, and thus can be neglected, since the first two terms on the right-hand side of (89) are $\mathcal{O}_P(P)$.

The average of the replicated partition function, after the three disorder averages over data, GEP noise and random features, reads

$$\overline{\mathcal{Z}^n} = \int\prod_c\left(d\boldsymbol{\mu}_c\,\frac{d\widehat{\boldsymbol{\mu}}_c}{(2\pi)^P}\right)\,e^{-\frac{1}{2N}\sum_{c,c'}\boldsymbol{m}_c^\intercal\boldsymbol{m}_{c'}\,\widehat{\boldsymbol{\mu}}_c^\intercal\widehat{\boldsymbol{\mu}}_{c'}+i\sum_c\widehat{\boldsymbol{\mu}}_c^\intercal\boldsymbol{\mu}_c}$$

$$\times\int\prod_c\left(d\boldsymbol{\Omega}_c\,\frac{d\widehat{\boldsymbol{\Omega}}_c}{(2\pi)^{nN}}\right)e^{-\frac{P\Delta}{2Ne^{-2t}}\sum_c\sum_a\frac{b_c}{\sigma_c^2}\boldsymbol{\omega}_{a;c}^\intercal\boldsymbol{\omega}_{a;c}+i\sqrt{P}\sum_c\sum_a\widehat{\boldsymbol{\omega}}_{a;c}^\intercal\boldsymbol{\omega}_{a;c}} \tag{90}$$

$$\times\int d\boldsymbol{\Psi}e^{-\frac{1}{2}\sum_c b_c\sum_a\boldsymbol{\psi}_a^\intercal(\operatorname{diag}(\boldsymbol{\varsigma}_c)-z\boldsymbol{I}_P)\boldsymbol{\psi}_a}e^{-\frac{M}{2}\sum_c b_c\log\det\left(\boldsymbol{I}_N+\frac{\chi_p}{M}\boldsymbol{J}_c'\right)}$$

$$\times e^{-\frac{M}{2}\sum_c b_c\log\det\left(I_P+\frac{1}{M}\operatorname{diag}(\boldsymbol{h}_c)\boldsymbol{\mathcal{J}}_c\operatorname{diag}(\boldsymbol{h}_c)\right)-\frac{1}{2}\sum_{c,c'}\sum_{ab}(\boldsymbol{\gamma}_c\odot\boldsymbol{\psi}_a)^\intercal(\boldsymbol{\gamma}_{c'}\odot\boldsymbol{\psi}_b)\,\widehat{\boldsymbol{\omega}}_{a;c}^\intercal\widehat{\boldsymbol{\omega}}_{b,c'}}$$

Average over the auxiliary fields. We can perform the Gaussian integrations with respect to the $\widehat{\boldsymbol{\omega}}_{a;c}$ and the $\widehat{\boldsymbol{\mu}}_c$: we start with the integral over the $\widehat{\boldsymbol{\omega}}_{a;c}$:

$$\int \frac{d\widehat{\boldsymbol{\Omega}}_c}{(2\pi)^{CnN}}\, e^{-\frac{1}{2}\sum_{cc'}\sum_{ab} K_{ab}^{cc'}\widehat{\boldsymbol{\omega}}_{a;c}^{\intercal}\widehat{\boldsymbol{\omega}}_{b;c'} - \sum_a\sum_c i\boldsymbol{\omega}_{a;c}^{\intercal}\widehat{\boldsymbol{\omega}}_{a;c}} =$$

$$= \frac{1}{(2\pi)^{CnN/2}} e^{-\frac{1}{2}\sum_{cc'}\sum_{ab}(\mathbf{K}^{-1})_{ab}^{cc'}\boldsymbol{\omega}_{a;c}^{\intercal}\boldsymbol{\omega}_{b;c'} - \frac{N}{2}\log\det\mathbf{K}}, \tag{91}$$

$$K_{ab}^{cc'} = \frac{1}{P}\operatorname{Tr}\left(\boldsymbol{\gamma}_c\boldsymbol{\gamma}_{c'}^{\intercal}\odot\boldsymbol{\psi}_a\boldsymbol{\psi}_b^{\intercal}\right).$$

Then, the integration over the $\widehat{\boldsymbol{\mu}}_c$ yields the probability density function (PDF) of the $\boldsymbol{\mu}_c$

$$\int \frac{d\widehat{\boldsymbol{\mu}_c}}{(2\pi)^{P/2}} e^{-\frac{1}{2}\sum_{cc'} r_{cc'}\widehat{\boldsymbol{\mu}}_c^{\intercal}\widehat{\boldsymbol{\mu}}_{c'} + i\sum_c \widehat{\boldsymbol{\mu}}_c^{\intercal}\boldsymbol{\mu}_c} = \frac{e^{-\frac{1}{2}\sum_{cc'}(\mathbf{r}^{-1})_{cc'}\boldsymbol{\mu}_c^{\intercal}\boldsymbol{\mu}_{c'}}}{(2\pi\det\mathbf{r})^{P/2}}, \tag{92}$$

$$r_{cc'} = \frac{1}{N}\mathbf{m}_c^{\intercal}\mathbf{m}_{c'}.$$

Thus, after these integrations the partition function reads

$$\begin{aligned}\overline{\mathcal{Z}^n} = &\int\prod_c \frac{d\boldsymbol{\mu}_c}{(2\pi)^{P/2}}\, e^{-\frac{1}{2}\sum_{c,c'}(\mathbf{r}^{-1})_{cc'}\,\boldsymbol{\mu}_c^{\intercal}\boldsymbol{\mu}_{c'} - \frac{1}{2}\log\det\mathbf{r}}\\ &\times\int\prod_c d\boldsymbol{\Omega}_c e^{-\frac{\chi_p\Delta}{2e^{-2t}}\sum_c\sum_a \frac{b_c}{\sigma_c^2}\boldsymbol{\omega}_{a;c}^{\intercal}\boldsymbol{\omega}_{a;c}}\\ \times\int d\boldsymbol{\Psi} e^{-\frac{1}{2}\sum_c b_c\sum_a \boldsymbol{\psi}_a^{\intercal}(\operatorname{diag}(\boldsymbol{\varsigma}_c)-z\boldsymbol{I}_P)\boldsymbol{\psi}_a - \frac{1}{2}\sum_{ab}\sum_{cc'}(\mathbf{K}(\boldsymbol{\Psi})^{-1})_{ab}^{cc'}\boldsymbol{\omega}_{a;c}^{\intercal}\boldsymbol{\omega}_{b;c'}}\\ \times e^{-\frac{N}{2}\log\det\mathbf{K}(\boldsymbol{\Psi}) - \frac{M}{2}\sum_c b_c\log\det\left(\mathbf{I}_N + \frac{\chi_p}{M}\boldsymbol{J}_c'\right) - \frac{M}{2}\sum_c b_c\log\det\left(\mathbf{I}_P + \frac{1}{M}\operatorname{diag}(\boldsymbol{h}_c)\boldsymbol{\mathcal{J}}_c\operatorname{diag}(\boldsymbol{h}_c)\right)}.\end{aligned} \tag{93}$$

Note that all the dependence on the projected centroids $\boldsymbol{\mu}_c$ is in the GEP Hermite coefficients, which features in the quantities $\boldsymbol{\varsigma}_c, \boldsymbol{h}_c, \mathbf{J}_c, \mathcal{J}_c$.

The replicated order parameters. We now introduce the order parameters of the problem: they are the following resolvent functions

$$g_{ab}^{(\Psi)} = \frac{1}{P}\operatorname{Tr}\boldsymbol{\psi}_a\boldsymbol{\psi}_b^{\intercal} \qquad g_{a;c}^{(\Psi,\varsigma)} = \frac{1}{P}\operatorname{Tr}\operatorname{diag}(\boldsymbol{\varsigma}_c)\boldsymbol{\psi}_a\boldsymbol{\psi}_a^{\intercal} \qquad g_{ab;cc'}^{(\Psi,\gamma)} = \frac{1}{P}\operatorname{Tr}\boldsymbol{\gamma}_c\boldsymbol{\gamma}_{c'}^{\intercal}\odot\boldsymbol{\psi}_a\boldsymbol{\psi}_b^{\intercal}$$

$$g_{a;c}^{(\Psi,h)} = \frac{1}{P}\operatorname{Tr}\boldsymbol{h}_c\boldsymbol{h}_c^{\intercal}\odot\boldsymbol{\psi}_a\boldsymbol{\psi}_a^{\intercal} \qquad\qquad g_{ab;cc'}^{(\Omega)} = \frac{1}{N}\operatorname{Tr}\boldsymbol{\omega}_{a;c}\boldsymbol{\omega}_{b,c'}^{\intercal}. \tag{94}$$

We enforce them in (93) through delta functions:

$$1 = \int d\{g_{ab}^{(\Psi)}\}\frac{d\{\widehat{g}_{ab}^{(\Psi)}\}}{(2\pi)^{n^2}} e^{iP\sum_{ab}\widehat{g}_{ab}^{(\Psi)}g_{ab}^{(\Psi)} - i\sum_{ab}\widehat{g}_{ab}^{(\Psi)}\boldsymbol{\psi}_a^{\intercal}\boldsymbol{\psi}_b}$$

$$1 = \int d\{g_{a;c}^{(\Psi,\varsigma)}\}\frac{d\{\widehat{g}_{a;c}^{(\Psi,\varsigma)}\}}{(2\pi)^{nC}} e^{iP\sum_a\sum_c\widehat{g}_{a;c}^{(\Psi,\varsigma)}g_{a;c}^{(\Psi,\varsigma)} - i\sum_a\sum_c\widehat{g}_{a;c}^{(\Psi,\varsigma)}\boldsymbol{\psi}_a^{\intercal}\operatorname{diag}(\boldsymbol{\varsigma}_c)\boldsymbol{\psi}_a}$$

$$1 = \int d\{g^{(\Psi,\gamma)}_{ab;cc'}\} \frac{d\{\widehat{g}^{(\Psi,\gamma)}_{ab,cc'}\}}{(2\pi)^{n^2C^2}} e^{iP\sum_{ab}\sum_{cc'}\widehat{g}^{(\Psi,\gamma)}_{ab;cc'}g^{(\Psi,\gamma)}_{ab;cc'} - i\sum_{ab}\sum_{cc'}\widehat{g}^{(\Psi,\gamma)}_{ab;cc'}\boldsymbol{\psi}^\intercal_a\boldsymbol{\gamma}_c\boldsymbol{\gamma}^\intercal_{c'}\boldsymbol{\psi}_b}$$

$$1 = \int d\{g^{(\Psi,h)}_{a;c}\} \frac{d\{\widehat{g}^{(\Psi,h)}_{a;c}\}}{(2\pi)^{nC}} e^{iP\sum_{a}\sum_{c}\widehat{g}^{(\Psi,h)}_{a;c}g^{(\Psi,h)}_{a;c} - i\sum_{a}\sum_{c}\widehat{g}^{(\Psi,h)}_{a;c}\boldsymbol{\psi}^\intercal_a\boldsymbol{h}_c\boldsymbol{h}^\intercal_c\boldsymbol{\psi}_a}$$

$$1 = \int d\{g^{(\Psi,\Omega)}_{ab;cc'}\} \frac{d\{\widehat{g}^{(\Psi,\Omega)}_{ab,cc'}\}}{(2\pi)^{n^2C^2}} e^{iN\sum_{ab}\sum_{cc'}\widehat{g}^{(\Psi,\Omega)}_{ab;cc'}g^{(\Psi,\Omega)}_{ab;cc'} - i\sum_{ab}\sum_{cc'}\widehat{g}^{(\Psi,\Omega)}_{ab;cc'}\boldsymbol{\omega}^\intercal_{a;c}\boldsymbol{\omega}_{b,c'}}$$

Then, we can integrate over the $\boldsymbol{\Psi}$ and the $\boldsymbol{\Omega}_c$ [2]:

$$\int d\boldsymbol{\Psi} e^{-i\sum_{ab}\boldsymbol{\psi}^\intercal_a\left[\widehat{g}^{(\Psi)}_{ab}\mathbf{I}_P + \delta_{ab}\sum_c \widehat{g}^{(\Psi,\varsigma)}_{a;c}\operatorname{diag}(\boldsymbol{\varsigma}_c) + \sum_{cc'}\widehat{g}^{(\Psi,\gamma)}_{ab;cc'}\operatorname{diag}(\boldsymbol{\gamma}_c\odot\boldsymbol{\gamma}_{c'}) + \delta_{ab}\sum_c\widehat{g}^{(\Psi,h)}_{a;c}\operatorname{diag}(\boldsymbol{h}^{\odot 2}_c)\right]\boldsymbol{\psi}_b}$$

$$\propto e^{-\frac{1}{2}\log\det\left(i\widehat{\mathbf{G}}^{(\Psi)}\otimes\mathbf{I}_P + \sum_c i\widehat{\boldsymbol{G}}^{(\Psi,\varsigma)}_c\otimes\operatorname{diag}(\boldsymbol{\varsigma}_c) + \sum_{cc'} i\widehat{\boldsymbol{G}}^{(\Psi,\gamma)}_{cc'}\otimes\operatorname{diag}(\boldsymbol{\gamma}_c\odot\boldsymbol{\gamma}_{c'}) + \sum_c i\widehat{\boldsymbol{G}}^{(\Psi,h)}_c\otimes\operatorname{diag}(\boldsymbol{h}^{\odot 2}_c)\right)}$$

$$\int d\{\boldsymbol{\Omega}_c\}\, e^{-i\sum_{ab}\sum_{cc'}\widehat{g}^{(\Omega)}_{ab;cc'}\boldsymbol{\omega}^\intercal_{a;b}\boldsymbol{\omega}_{b;c'}} \propto e^{-\frac{N}{2}\log\det\left(i\widehat{\boldsymbol{G}}^{(\Omega)}\right)}$$

The Replica-Symmetric ansatz and interpretation of the order parameters. We impose a Replica-Symmetric (RS) ansatz [3]

$$g^{(\Psi)}_{ab} = \begin{cases} g^{(\Psi)} & a = b \\ 0 & a \neq b \end{cases} \qquad \widehat{g}^{(\Psi)}_{ab} = \begin{cases} \widehat{g}^{(\Psi)} & a = b \\ 0 & a \neq b \end{cases} \tag{95}$$

$$g^{(\Psi,\varsigma)}_{a;c} = g^{(\Psi,\varsigma)}_c \qquad \widehat{g}^{(\Psi,\varsigma)}_{a;c} = \widehat{g}^{(\Psi,\varsigma)}_c \tag{96}$$

$$g^{(\Psi,\gamma)}_{ab;cc'} = \begin{cases} g^{(\Psi,\gamma)}_{cc'} & a = b \\ 0 & a \neq b \end{cases} \qquad \widehat{g}^{(\Psi,\gamma)}_{ab;cc'} = \begin{cases} \widehat{g}^{(\Psi,\gamma)}_{cc'} & a = b \\ 0 & a \neq b \end{cases} \tag{97}$$

$$g^{(\Psi,h)}_{a;c} = g^{(\Psi,h)}_c \qquad \widehat{g}^{(h)}_{a;c} = \widehat{g}^{(\Psi,h)}_c \tag{98}$$

$$g^{(\Omega)}_{ab;cc'} = \begin{cases} g^{(\Omega)}_{cc'} & a = b \\ 0 & a \neq b \end{cases} \qquad \widehat{g}^{(\Omega)}_{ab;cc'} = \begin{cases} \widehat{g}^{(\Omega)}_{cc'} & a = b \\ 0 & a \neq b \end{cases} \tag{99}$$

The order parameters (94) represent the statistics sufficient to characterize the spectral properties of the matrix $\widehat{\mathbf{U}}$: the $g^{(\Psi)}$ represents the resolvent function of $\widehat{\mathbf{U}}$ itself (see Eq. (106) later in the text), while the other order parameters capture the influence of different elements of the RF model on the spectrum of $\widehat{U}$. The $g^{(\Omega)}_{cc'}$ are resolvent functions of the projection of $\widehat{\mathbf{U}}$ in data space: one can verify that

$$g^{(\Omega)}_{cc'} = \frac{1}{NP}\operatorname{Tr}\left[\mathbf{W}^\intercal\operatorname{diag}(\boldsymbol{\gamma}_c)(\widehat{\mathbf{U}} - z)^{-1}\operatorname{diag}(\boldsymbol{\gamma}_{c'})\mathbf{W}\right]. \tag{100}$$

They capture the correlations of the classes observed by the model, projected back into the data space. The resolvents $g^{(\Psi,h)}_c, g^{(\Psi,\varsigma)}_c$ respectively capture the effect of GEP noise variances ($\boldsymbol{h}^{\odot 2}_c$) and feature variances ($\boldsymbol{\varsigma}_c$): one can check that

$$g^{(\Psi,h)}_c = \frac{1}{P}\operatorname{Tr}\left[\operatorname{diag}(\boldsymbol{h}_c)(\widehat{\mathbf{U}} - z)^{-1}\operatorname{diag}(\boldsymbol{h}_c)\right] \tag{101}$$

[2] We ignore factors that are not exponential in system size.

[3] Random matrix theories are notoriously RS stable [43]. The RS diagonal ansatz for the replicas is usually used ab initio, but it can be easily verified starting from a more generic RS ansatz.

$$g_c^{(\Psi,\varsigma)} = \frac{1}{P}\,\mathrm{Tr}\left[\mathrm{diag}(\boldsymbol{\varsigma}_c^{\odot 1/2})(\widehat{\mathbf{U}}-z)^{-1}\,\mathrm{diag}(\boldsymbol{\varsigma}_c^{\odot 1/2})\right]. \tag{102}$$

Finally, the $g_{cc'}^{(\Psi,\gamma)}$ express class correlations observed by the model in feature space:

$$g_{cc'}^{(\Psi,\gamma)} = \frac{1}{P}\,\mathrm{Tr}\left[\mathrm{diag}(\boldsymbol{\gamma}_c)(\widehat{\mathbf{U}}-z)^{-1}\,\mathrm{diag}(\boldsymbol{\gamma}_{c'})\right]. \tag{103}$$

When centroids are extensive, these effects are heterogeneous in feature space: the model displays a different expressivity among the different hidden neurons. Essentially, centroids make neuron activations statistically correlated. In contrast, in the limit of subextensive centroids studied later in Appendix H.3, the model acts homogeneously on the different hidden neurons: in this limit, one has $\boldsymbol{\gamma}_c \to \gamma_c \mathbf{1}_P$ and similarly for the other GEP vectors.

Applying the RS diagonal ansatz. After clarifying the meaning of the order parameters, we proceed by simplifying all the terms in Eq. (93), taking into account the previous integrations over the $\boldsymbol{\Psi}, \boldsymbol{\Omega}$:

$$\log\det\Bigg(i\widehat{\mathbf{G}}^{(\Psi)}\otimes\mathbf{I}_P + \sum_c i\widehat{\boldsymbol{G}}_c^{(\Psi,\varsigma)}\otimes\,\mathrm{diag}(\boldsymbol{\varsigma}_c)$$

$$+\sum_{cc'} i\widehat{\boldsymbol{G}}_{cc'}^{(\Psi,\gamma)}\otimes\,\mathrm{diag}(\boldsymbol{\gamma}_c\odot\boldsymbol{\gamma}_{c'}) + \sum_c i\widehat{\boldsymbol{G}}_c^{(\Psi,h)}\otimes\,\mathrm{diag}(\boldsymbol{h}_c^{\odot 2})\Bigg)\simeq$$

$$n\log\det\left(i\widehat{g}_\Psi\mathbf{I}_P + \sum_c i\widehat{g}_c^{(\Psi,\varsigma)}\,\mathrm{diag}(\boldsymbol{\varsigma}_c) + \sum_{cc'} i\widehat{g}_{cc'}^{(\Psi,\gamma)}\,\mathrm{diag}(\boldsymbol{\gamma}_c\odot\boldsymbol{\gamma}_{c'}) + \sum_c i\widehat{g}_c^{(\Psi,h)}\,\mathrm{diag}(\boldsymbol{h}_c^{\odot 2})\right) =$$

$$n\sum_{p=1}^{P}\log\left(i\widehat{g}_\Psi + \sum_c i\widehat{g}_c^{(\Psi,\varsigma)}\varsigma_c^{(p)} + \sum_{cc'} i\widehat{g}_{cc'}^{(\Psi,\gamma)}\gamma_c^{(p)}\gamma_{c'}^{(p)} + \sum_c i\widehat{g}_c^{(\Psi,h)}(h_c^{(p)})^2\right)$$

$$\log\det\left(i\widehat{\boldsymbol{G}}^{(\Omega)}\right)\simeq n\log\det i\{\widehat{g}_{cc'}^{(\Omega)}\}_{cc'}$$

$$\log\det\left(\boldsymbol{I}_N+\frac{\chi_p}{\chi_m}\boldsymbol{J}_c\right)\simeq n\log\left(1+\frac{\chi_p}{\chi_m}g_{cc}^{(\Omega)}\right)$$

$$\log\det\left(\mathbf{I}_P+\frac{1}{\chi_m}\,\mathrm{diag}(\boldsymbol{h}_c^{\odot 2})\boldsymbol{\mathcal{J}}_c\right)\simeq n\log\left[1+\frac{\chi_p}{\chi_m}\left(1-\frac{\chi_p}{\chi_m}\frac{g_{cc}^{(\Omega)}}{1+\frac{\chi_p}{\chi_m}g_{cc}^{(\Omega)}}\right)g_{a;c}^{(\Psi,h)}\right]$$

$$\mathrm{K}_{ab}^{cc'}=\delta_{ab}g_{cc'}^{(\Psi,\gamma)}\Longrightarrow(\mathbf{K}^{-1})_{ab}^{cc'}=\delta_{ab}\,(\boldsymbol{g}_{(\Psi,\gamma)}^{-1})_{cc'}$$

$$\frac{1}{2\chi_p}\sum_{ab}\sum_{cc'}(\mathbf{K}(\boldsymbol{g}^{(\Psi,\gamma)})^{-1})_{ab}^{cc'}\,g_{ab;cc'}^{(\Omega)}\simeq\frac{n}{2\chi_p}\sum_{cc'}(\boldsymbol{g}_{(\Psi,\gamma)}^{-1})_{cc'}g_{cc'}^{(\Omega)}$$

$$\frac{1}{2\chi_p}\log\det\mathbf{K}\simeq\frac{n}{2\chi_p}\log\det\boldsymbol{g}^{(\Psi,\gamma)}$$

Here and in the following lines, the symbols $\boldsymbol{g}^{(\Psi,\gamma)}$ and $\boldsymbol{g}^{(\Omega)}$ indicate the matrices $g_{cc'}^{(\Psi,\gamma)}$, $g_{cc'}^{(\Omega)}$ for $c,c'=1,\dots,C$. Collecting everything, we can finally write

$$\overline{\mathcal{Z}^n}\propto\mathbb{E}_{\{\mu_c\}}\left[\int d\{\boldsymbol{g}^{(\mathrm{all})}\}d\{\widehat{\boldsymbol{g}}^{(\mathrm{all})}\}e^{nP\mathcal{F}\left(\{\boldsymbol{g}^{(\mathrm{all})}\},\widehat{\boldsymbol{g}}^{(\mathrm{all})}\}\right)}\right]\propto\mathbb{E}_{\{\mu_c\}}\left[e^{nP\mathcal{F}\left(\{\boldsymbol{g}_*^{(\mathrm{all})}\},\widehat{\boldsymbol{g}}_*^{(\mathrm{all})}\}\right)}\right] \tag{104}$$

The compact notation $\{\boldsymbol{g}^{(\mathrm{all})}\}, \{\widehat{\boldsymbol{g}}^{(\mathrm{all})}\}$ identifies all the order parameters and their conjugate parameters. The action $\mathcal{F}$, in its final form, reads

$$\begin{aligned}
\mathcal{F}\left(\{\boldsymbol{g}^{(\mathrm{all})}, \widehat{\boldsymbol{g}}^{(\mathrm{all})}\}\right) &= i\widehat{g}_\Psi g_\Psi + \sum_c i\widehat{g}_c^{(\Psi,\varsigma)} g_c^{(\Psi,\varsigma)} + \sum_{cc'} i\widehat{g}_{cc'}^{(\Psi,\gamma)} g_{cc'}^{(\Psi,\gamma)} \\
&+ \sum_c i\widehat{g}_c^{(\Psi,h)} g_c^{(\Psi,h)} + \frac{1}{\chi_p}\sum_{cc'} i\widehat{g}_{cc'}^{(\Omega)} g_{cc'}^{(\Omega)} - \frac{1}{2\chi_p}\log\det i\widehat{\boldsymbol{g}}_\Omega \\
&- \frac{1}{2P}\sum_{p=1}^{P}\log\left(i\widehat{g}_\Psi + \sum_c i\widehat{g}_c^{(\Psi,\varsigma)}\varsigma_c^{(p)} + \sum_{cc'} i\widehat{g}_{cc'}^{(\Psi,\gamma)}\gamma_c^{(p)}\gamma_{c'}^{(p)} + \sum_c i\widehat{g}_c^{(\Psi,h)}(h_c^{(p)})^2\right) \\
&- \frac{\Delta}{2e^{-2t}}\sum_c \frac{b_c}{\sigma_c^2} g_{cc}^{(\Omega)} - \frac{1}{2\chi_p}\sum_{cc'}(\boldsymbol{g}_{(\Psi,\gamma)}^{-1})_{cc'} g_{cc'}^{(\Omega)} \\
&- \frac{1}{2\chi_p}\log\det \boldsymbol{g}^{(\Psi,\gamma)} - \frac{1}{2}\sum_c b_c\left(g_c^{(\Psi,\varsigma)} - z\; g^{(\Psi)}\right) \\
&- \frac{\chi_m}{2\chi_p}\sum_c b_c \log\left(1 + \frac{\chi_p}{\chi_m} g_{cc}^{(\Omega)} + \frac{\chi_p}{\chi_m} g_c^{(\Psi,h)}\right)
\end{aligned} \tag{105}$$

We finally use (81) to get

$$\begin{aligned}
g(z) = 2\frac{\partial}{\partial z}\lim_{P\to\infty}\frac{1}{P}\mathbb{E}_{\mathbf{X},\mathbf{W}}\left[\log \mathcal{Z}_U\right] = \\
2\frac{\partial}{\partial z}\lim_{n\to 0}\lim_{P\to\infty}\frac{1}{P}\frac{\mathbb{E}_{\{\boldsymbol{\mu_c}\}}\left[e^{nP\mathcal{F}}\right]-1}{n} = 2\frac{\partial}{\partial z}\mathbb{E}_{\{\mu_c\}}\left[\mathcal{F}\right] = g_\Psi^*(z).
\end{aligned} \tag{106}$$

Derivation of the spectral equations. The order parameter g_Ψ^* maximizing the action (105) is the resolvent function that we wish to determine. This quantity depends on all the other resolvents appearing in (105): to understand how, we need to write the saddle point equations for (105), whose solutions are the resolvents yielding the maximum of (105). For brevity of notation, we omit the symbol $*$ on the resolvents, previously used to indicate solutions of saddle point equations. We start with those for the conjugate order parameters:

$$\frac{\partial\mathcal{F}}{\partial g_\Psi} = 0 \Longrightarrow i\widehat{g}_\Psi = -\frac{z}{2} \tag{107}$$

$$\frac{\partial\mathcal{F}}{\partial g_c^{(\Psi,\varsigma)}} = 0 \Longrightarrow i\widehat{g}_c^{(\Psi,\varsigma)} = \frac{b_c}{2} \tag{108}$$

$$\frac{\partial\mathcal{F}}{\partial g_c^{(\Psi,h)}} = 0 \Longrightarrow i\widehat{g}_c^{(\Psi,h)} = \frac{b_c}{2}\frac{1}{1+\frac{\chi_p}{\chi_m}g_{cc}^{(\Omega)} + \frac{\chi_p}{\chi_m}g_c^{(\Psi,h)}} \tag{109}$$

$$\begin{aligned}
\frac{\partial\mathcal{F}}{\partial \boldsymbol{g}_{(\Psi,\gamma)}} = 0 \Longrightarrow i\widehat{\boldsymbol{g}}_{(\Psi,\gamma)} &= \frac{1}{2\chi_p}\frac{\partial}{\partial \boldsymbol{g}_{(\Psi,\gamma)}}\mathrm{Tr}\left(\boldsymbol{g}_{(\Psi,\gamma)}^{-1}\boldsymbol{g}_{(\Omega)}^{\intercal}\right) + \frac{1}{2\chi_p}\frac{1}{\det \boldsymbol{g}^{(\Psi,\gamma)}}\frac{\partial \det \boldsymbol{g}^{(\Psi,\gamma)}}{\partial \boldsymbol{g}_{(\Psi,\gamma)}} \\
&= -\frac{1}{2\chi_p}\boldsymbol{g}_{(\Psi,\gamma)}^{-1}\boldsymbol{g}_\Omega\boldsymbol{g}_{(\Psi,\gamma)}^{-1} + \frac{1}{2\chi_p}\boldsymbol{g}_{(\Psi,\gamma)}^{-1}
\end{aligned} \tag{110}$$

$$\frac{\partial \mathcal{F}}{\partial g_{cc}^{(\Omega)}} = 0 \Longrightarrow i\widehat{g}_{cc}^{(\Omega)} = \frac{1}{2}(\boldsymbol{g}_{(\Psi,\gamma)}^{-1})_{cc} + \frac{b_c}{2}\frac{\chi_p}{1+\frac{\chi_p}{\chi_m}g_{cc}^{(\Omega)}+\frac{\chi_p}{\chi_m}g_c^{(\Psi,h)}} + \frac{\Delta\chi_p}{2e^{-2t}}\frac{b_c}{\sigma_c^2} \tag{111}$$

$$\frac{\partial \mathcal{F}}{\partial g_{cc'}^{(\Omega)}} = 0 \Longrightarrow i\widehat{g}_{cc'}^{(\Omega)} = \frac{1}{2}(\boldsymbol{g}_{(\Psi,\gamma)}^{-1})_{cc'}. \tag{112}$$

Second, we write the saddle point equations for the order parameters:

$$\frac{\partial \mathcal{F}}{\partial(i\widehat{g}_\Psi)} = 0$$

$$\Longrightarrow g_\Psi = \frac{1}{2P}\sum_{p=1}^{P}\frac{1}{i\widehat{g}_\Psi + \sum_c i\widehat{g}_c^{(\Psi,\varsigma)}\varsigma_c^{(p)} + \sum_{cc'} i\widehat{g}_{cc'}^{(\Psi,\gamma)}\gamma_c^{(p)}\gamma_{c'}^{(p)} + \sum_c i\widehat{g}_c^{(\Psi,h)}(h_c^{(p)})^2} \tag{113}$$

$$\frac{\partial \mathcal{F}}{\partial(i\widehat{g}_c^{(\Psi,\varsigma)})} = 0$$

$$\Longrightarrow g_c^{(\Psi,\varsigma)} = \frac{1}{2P}\sum_{p=1}^{P}\frac{\varsigma_c^{(p)}}{i\widehat{g}_\Psi + \sum_c i\widehat{g}_c^{(\Psi,\varsigma)}\varsigma_c^{(p)} + \sum_{cc'} i\widehat{g}_{cc'}^{(\Psi,\gamma)}\gamma_c^{(p)}\gamma_{c'}^{(p)} + \sum_c i\widehat{g}_c^{(\Psi,h)}(h_c^{(p)})^2} \tag{114}$$

$$\frac{\partial \mathcal{F}}{\partial(i\widehat{g}_c^{(\Psi,h)})} = 0$$

$$\Longrightarrow g_c^{(\Psi,h)} = \frac{1}{2P}\sum_{p=1}^{P}\frac{(h_c^{(p)})^2}{i\widehat{g}_\Psi + \sum_c i\widehat{g}_c^{(\Psi,\varsigma)}\varsigma_c^{(p)} + \sum_{cc'} i\widehat{g}_{cc'}^{(\Psi,\gamma)}\gamma_c^{(p)}\gamma_{c'}^{(p)} + \sum_c i\widehat{g}_c^{(\Psi,h)}(h_c^{(p)})^2} \tag{115}$$

$$\frac{\partial \mathcal{F}}{\partial(i\widehat{g}_{cc'}^{(\Psi,\gamma)})} = 0$$

$$\Longrightarrow g_{cc'}^{(\Psi,\gamma)} = \frac{1}{2P}\sum_{p=1}^{P}\frac{\gamma_c^{(p)}\gamma_{c'}^{(p)}}{i\widehat{g}_\Psi + \sum_c i\widehat{g}_c^{(\Psi,\varsigma)}\varsigma_c^{(p)} + \sum_{cc'} i\widehat{g}_{cc'}^{(\Psi,\gamma)}\gamma_c^{(p)}\gamma_{c'}^{(p)} + \sum_c i\widehat{g}_c^{(\Psi,h)}(h_c^{(p)})^2} \tag{116}$$

$$\frac{\partial \mathcal{F}}{\partial(i\widehat{g}_{cc'}^{(\Omega)})} = 0 \Longrightarrow g_{cc'}^{(\Omega)} = \frac{1}{2}((i\widehat{\boldsymbol{g}}_\Omega)^{-1})_{cc'}. \tag{117}$$

H.2 Spectral equations for extensive centroids

We can finally write the equations necessary to solve for the spectrum of $\widehat{\mathbf{U}}$ by combining Eqs. (107), (109), (110), (111), (112) with Eqs. (113), (115), (116), (117). Note that Eqs. (108) and (114) are not necessary to determine the resolvent function $g(z) \equiv g_\Psi(z)$, since $i\widehat{g}_c^{(\Psi,\varsigma)}$ is a constant (Eq. (108)), independent of all other resolvents. For the sake of compactness, we simplify the notation of the auxiliary resolvents and define the functions

$$\mathcal{D}(\{\mu_c\}) = \sum_c b_c \varsigma_c(\mu_c) \tag{118}$$

$$\mathcal{R}_z(\{g_c^{(h)}(z), g_{cc'}^{(\gamma)}(z), g_{cc'}^{(\Omega)}(z)\}, \{\mu_c\}) = -\frac{2}{\chi_p}\sum_{cc'} \mathrm{y}_{cc'}(\boldsymbol{g}^{(\Psi,\gamma)}, \boldsymbol{g}^{(\Omega)})\ \gamma_c(\mu_c)\gamma_{c'}(\mu_{c'}) - \sum_c \frac{b_c h_c^2(\mu_c)}{1+\frac{\chi_p}{\chi_m}g_{cc}^{(\Omega)}+\frac{\chi_p}{\chi_m}g_c^{(\Psi,h)}} \tag{119}$$

for $\mu_c \in \mathbb{R}, c = 1, \ldots, C$. In Eq. (119), we renamed $\mathrm{y}_{cc'} = i\chi_p \widehat{g}_{cc'}^{(\Psi,\gamma)}, \forall c, c' = 1, \ldots, C$. Moreover, in (118) and (119) we introduced the GEP functions

$$\varsigma_c(\mu) = \widetilde{\beta}_c(\mu) - \beta_c(\mu) - \frac{\Delta}{\sigma_c^2 e^{-2t}}\gamma_c^2(\mu)$$

$$h_c^2(\mu) = \beta_c(\mu) - \alpha_c^2(\mu) - \gamma_c^2(\mu)$$

with

$$\alpha_c(\mu) = \mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\phi\left(\Gamma_c(t)x + e^{-t}\mu\right)\right] \tag{120}$$

$$\widetilde{\beta}_c(\mu) = \mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\phi^2\left(\Gamma_c(t)x + e^{-t}\mu\right)\right]$$

$$\beta_c(\mu) = \mathbb{E}_{x_1,x_2\sim\mathcal{N}(0,\boldsymbol{\Sigma}_c(t))}\left[\phi\left(\Gamma_c(t)x_1 + e^{-t}\mu\right)\phi\left(\Gamma_c(t)x_2 + e^{-t}\mu\right)\right]$$

$$\gamma_c(\mu) = \frac{\sigma_c e^{-t}}{\Gamma_c(t)}\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[x\ \phi(\Gamma_c(t)x + e^{-t}\mu)\right].$$

Finally, let us use the notation

$$\mathbb{E}_{\{\mu_c\}}[\cdot] \equiv \int \frac{d\mu_1 \ldots d\mu_C}{(2\pi)^{C/2}\sqrt{\det \boldsymbol{r}}} e^{-\frac{1}{2}\sum_{cc'}\mu_c(\boldsymbol{r}^{-1})_{cc'}\mu_{c'}}(\cdot), \tag{121}$$

with $r_{cc'} = \frac{1}{N}\boldsymbol{m}_c \cdot \boldsymbol{m}_{c'}$. Then, in the limit $P \to \infty$ the resolvent $g(z)$ can be determined by solving the following system of equations for $k, k' = 1, \ldots, C$:

$$g_k^{(h)}(z) = \mathbb{E}_{\{\mu_c\}}\left[\frac{h_k^2(\mu_k)}{\mathcal{D}(\{\mu_c\}) - z - \mathcal{R}(\{g_c^{(h)}(z), g_{cc'}^{(\gamma)}(z), g_{cc'}^{(\Omega)}(z)\}, \{\mu_c\})}\right]$$

$$g_{kk'}^{(\gamma)}(z) = \mathbb{E}_{\{\mu_c\}}\left[\frac{\gamma_k(\mu_k)\gamma_{k'}(\mu_{k'})}{\mathcal{D}(\{\mu_c\}) - z - \mathcal{R}(\{g_c^{(h)}(z), g_{cc'}^{(\gamma)}(z), g_{cc'}^{(\Omega)}(z)\}, \{\mu_c\})}\right]$$

$$i\widehat{g}_{kk}^{(\Omega)} = \frac{1}{2}(\boldsymbol{g}_{(\gamma)}^{-1})_{kk} + \frac{b_k}{2}\frac{\chi_p}{1+\frac{\chi_p}{\chi_m}g_{kk}^{(\Omega)}+\frac{\chi_p}{\chi_m}g_k^{(h)}} + \frac{\Delta\chi_p}{2e^{-2t}}\frac{b_k}{\sigma_k^2} \tag{122}$$

$$i\widehat{g}_{kk'}^{(\Omega)} = \frac{1}{2}(\boldsymbol{g}_{(\gamma)}^{-1})_{kk'} \qquad k \neq k'$$

$$\boldsymbol{g}_{(\Omega)} = \frac{1}{2}(i\widehat{\boldsymbol{g}}_\Omega)^{-1}.$$

The resolvent $g(z)$ can be computed from the solution of (122) through the formula

$$\boxed{g(z) = \mathbb{E}_{\{\mu_c\}}\left[\frac{1}{\mathcal{D}(\{\mu_c\}) - z - \mathcal{R}(\{g_c^{(h)}(z), g_{cc'}^{(\gamma)}(z), g_{cc'}^{(\Omega)}(z)\}, \{\mu_c\})}\right].} \tag{123}$$

Equations like (122), usually in a much simpler form, appear whenever a random matrix ensemble is non-rotationally invariant: by inspection of (113) and (123), (119), (118), one can deduce that the diagonal entries of the resolvent matrix $\mathbf{G}(z)$ yielded by the replica computation in the limit $P \to \infty$ are

$$G_{pp}(z) = \frac{1}{\mathcal{D}(\mu_1^{(p)}, \ldots, \mu_C^{(p)}) - z - \mathcal{R}(\{g_c^{(h)}(z), g_{cc'}^{(\gamma)}(z), g_{cc'}^{(\Omega)}(z)\}, \mu_1^{(p)}, \ldots, \mu_C^{(p)}),} \tag{124}$$

where $p = 1, \ldots, P$. The diagonal entries of the resolvent are fluctuating quantities also in the high-dimensional limit, correlated to the statistics of the projected class centroids. Conversely, in rotationally invariant ensembles, like the Wishart ensemble, the diagonal entries of the resolvent concentrate around their mean in the high-dimensional limit.

(122) is a system of $C(C+2)$ complex equations[4], for given $z \in \mathbb{C}$ [5]. In the multiclass case $C > 2$, it is a rather difficult system to solve. In the special binary case, $C = 2$, it is possible to simplify some terms in (122): first, the explicit expression of the matrix inverse

$$\boldsymbol{g}_{(\Psi,\gamma)}^{-1} = \frac{1}{\det \boldsymbol{g}_{(\Psi,\gamma)}} \begin{pmatrix} g_{22}^{(\Psi,\gamma)} & -g_{12}^{(\Psi,\gamma)} \\ -g_{12}^{(\Psi,\gamma)} & g_{11}^{(\Psi,\gamma)} \end{pmatrix} \qquad \det \boldsymbol{g}_{(\Psi,\gamma)} = g_{11}^{(\Psi,\gamma)} g_{22}^{(\Psi,\gamma)} - (g_{12}^{(\Psi,\gamma)})^2 \tag{125}$$

we can write for $\mathbf{y}$:

$$\begin{aligned} \mathbf{y} = \frac{1}{2\chi_p} \frac{1}{g_{11}^{(\Psi,\gamma)} g_{22}^{(\Psi,\gamma)} - (g_{12}^{(\Psi,\gamma)})^2} \begin{pmatrix} g_{22}^{(\Psi,\gamma)} & -g_{12}^{(\Psi,\gamma)} \\ -g_{12}^{(\Psi,\gamma)} & g_{11}^{(\Psi,\gamma)} \end{pmatrix} \\ \times \left[\mathbf{I}_2 - \frac{1}{g_{11}^{(\Psi,\gamma)} g_{22}^{(\Psi,\gamma)} - (g_{12}^{(\Psi,\gamma)})^2} \begin{pmatrix} g_{22}^{(\Psi,\gamma)} & -g_{12}^{(\Psi,\gamma)} \\ -g_{12}^{(\Psi,\gamma)} & g_{11}^{(\Psi,\gamma)} \end{pmatrix} \begin{pmatrix} g_{11}^{(\Omega)} & g_{12}^{(\Omega)} \\ g_{12}^{(\Omega)} & g_{22}^{(\Omega)} \end{pmatrix} \right] \end{aligned} \tag{126}$$

[4]The total number of equations is actually $2C(C+2)$, since the system is solved for the real and imaginary parts of each of the resolvents in (122).

[5]In the high-dimensional limit, in regions of the spectrum with absolute continuous measure, the resolvent develops a branch cut singularity for $\mathrm{Im}\, z = 0$, rather than having first order poles on the eigenvalues, as happens for discrete spectra. In this situation, the equations that define the spectrum are well defined in the whole complex plane.

and unveil the equations for the entries of $\mathbf{g}_\Omega$, by explicitly writing the matrix inverse in (117) and using (111), (112)

$$
\begin{aligned}
g_{11}^{(\Omega)} &= \frac{l_{22}}{l_{11}l_{22} - [(\boldsymbol{g}^{-1}_{(\Psi,\gamma)})_{12}]^2} \\
g_{22}^{(\Omega)} &= \frac{l_{11}}{l_{11}l_{22} - [(\boldsymbol{g}^{-1}_{(\Psi,\gamma)})_{12}]^2} \\
g_{12}^{(\Omega)} &= -\frac{(\boldsymbol{g}^{-1}_{(\Psi,\gamma)})_{12}}{l_{11}l_{22} - [(\boldsymbol{g}^{-1}_{(\Psi,\gamma)})_{12}]^2}
\end{aligned}
\tag{127}
$$

with

$$
l_{cc} = \chi_p \left[\frac{\Delta b_c}{e^{-2t}\sigma_c^2} + \frac{1}{\chi_p}(\boldsymbol{g}^{-1}_{(\Psi,\gamma)})_{cc} + \frac{b_c}{1 + \frac{\chi_p}{\chi_m} g_{cc}^{(\Omega)} + \frac{\chi_p}{\chi_m} g_c^{(\Psi,h)}} \right].
$$

In the binary case, the system (122) counts 16 self-consistent real equations. We reserve the study of Eqs. (122) for future work.

H.3 Spectral equations for subextensive centroids

We consider the case of subextensive centroids, i.e. for any class c one has $\|\boldsymbol{\mu}_c\| = o_P(P)$: this setting represents datasets with centered classes. In this case, $P(\mu_1, \dots, \mu_C) \simeq \delta(\mu_1) \cdots \delta(\mu_C)$ and the problem can be considered relatively straightforward: defining for $c = 1, \dots, C$ the scalars

$$
\begin{aligned}
\widetilde{\beta}_c &= \mathbb{E}_z \left[\phi^2(\Gamma_c z)\right] \qquad\qquad \gamma_c = \mathbb{E}_z \left[z\, \phi(\Gamma_c z)\right] \\
\beta_c &= \mathbb{E}_{(z_1,z_2)\sim\mathcal{N}(0,\boldsymbol{\Sigma}_c(t))} \left[\phi(\Gamma_c z_1)\phi(\Gamma_c z_2)\right] \\
\varsigma_c &= \widetilde{\beta}_c - \beta_c - \frac{\Delta\gamma_c^2}{\sigma_c^2 e^{-2t}} \qquad h_c^2 = \beta_c - \gamma_c^2
\end{aligned}
\tag{128}
$$

the resolvent function $g(z) \equiv g_\Psi(z)$ is determined from the solution of the following system of saddle point equations:

$$
\boxed{
\begin{aligned}
\left(1 - \frac{1}{\chi_p}\right)\frac{1}{g_\Psi} &= -\frac{1}{\chi_p}\frac{g_\Omega}{g_\Psi^2} + \sum_c b_c\,(\varsigma_c - z) + \sum_c \frac{b_c h_c^2}{1 + \frac{\chi_p}{\chi_m}[\gamma_c^2 g_\Omega + h_c^2 g_\Psi]} \\
\frac{1}{\chi_p}\frac{1}{g_\Omega} &= \frac{\Delta}{e^{-2t}}\sum_c \frac{b_c}{\sigma_c^2}\gamma_c^2 + \frac{1}{\chi_p}\frac{1}{g_\Psi} + \sum_c \frac{b_c\gamma_c^2}{1 + \frac{\chi_p}{\chi_m}[\gamma_c^2 g_\Omega + h_c^2 g_\Psi]}
\end{aligned}
}
\tag{129}
$$

Eqs. (129) can be obtained as follows: first, write the action (105) in the limit of subextensive centroids, thus replacing the vector Hermite coefficients with their homogeneous counterparts in Eqs. (128), $\widetilde{\boldsymbol{\beta}}_c \to \widetilde{\beta}_c \mathbf{1}_P$, $\boldsymbol{\beta}_c \to \beta_c \mathbf{1}_P$, $\boldsymbol{\gamma}_c \to \gamma_c \mathbf{1}_P$ (note that, for an odd activation, the coefficient $\boldsymbol{\alpha}_c$ vanishes in this limit). Second, note that $g_c^{(\Psi,h)} \to h_c^2\, g_\Psi$, $g_c^{(\Psi,\varsigma)} \to \varsigma_c\, g_\Psi$, $g_{cc'}^{(\Psi,\gamma)} \to \gamma_c\gamma_{c'}\, g_\Psi$, $g_{cc'}^{(\Omega)} \to \gamma_c\gamma_{c'}\, g_\Omega$; to handle singular terms such as determinants and inverses of these last two

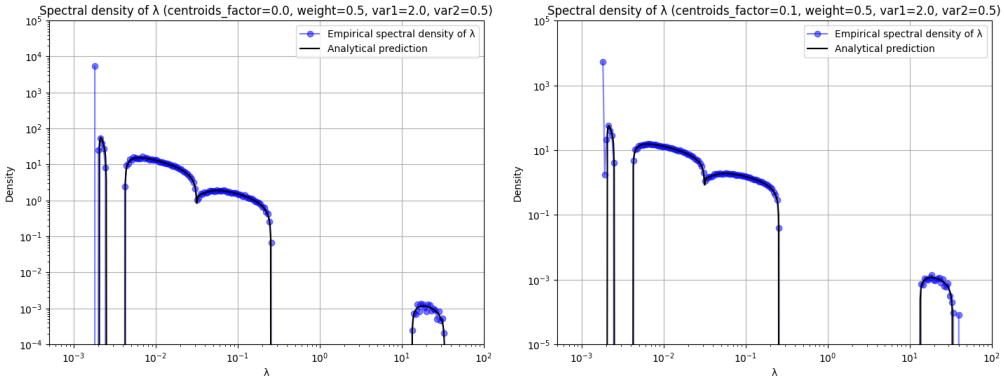


Figure 10: Test of Eqs. (129) for $t = 0.01$.

resolvents appearing in (105), set $g_{cc}^{(\Psi,\gamma)} = \gamma_c^2\, g_\Psi + \epsilon$, $g_{cc}^{(\Omega)} = \gamma_c^2\, g_\Omega + \epsilon$, expand and simplify. Then, derive the saddle point equations of the resulting action and obtain (129).

We remark that, while for extensive centroids $\widehat{\mathbf{U}}_{gep}$ in (10) is non-rotationally invariant, in the limit of subextensive centroids $\widehat{\mathbf{U}}_{gep}$ consists of the sum of different Wishart matrices and therefore, the resulting matrix ensemble is rotationally invariant: usually, in this case the spectral equations are algebraic, and this is also the case for (129). Finally, note that, in contrast with equations (122), in the case of subextensive centroids the total number of real equations to solve is always 4, for any number of classes C.

For $\sigma_1 = \cdots = \sigma_C = \sigma$, Eqs. (129) reduce to the result obtained in [8] for data sampled from a single gaussian.

In figure 10 we compare the solution of (129) in the $C = 2$ case with empirical histograms of the eigenvalues of $\widehat{\mathbf{U}}$, testing the case of zero centroid norm (left) and $\|\boldsymbol{m}_1\| = \|\boldsymbol{m}_2\| = 1$ (right): we find a perfect agreement between theory and simulations. We compute the theoretical spectral density from the solution of (129) using (80).

H.3.1 Equations for the edges

The equations for the spectral edges λ_* can be obtained from the spectral equations (129) by performing the following joint limits

$$\lim_{\lambda\to\lambda_*} \operatorname{Im} g_\Psi(\lambda) = \lim_{\lambda\to\lambda_*} \operatorname{Im} g_\Omega(\lambda) = 0, \qquad \lim_{\lambda\to\lambda_*} \frac{\operatorname{Im} g_\Omega(\lambda)}{\operatorname{Im} g_\Psi(\lambda)} \equiv \omega_* \in \mathbb{R}/\{0\}. \tag{130}$$

For any bulk of the spectrum, the lower and upper edges are defined, respectively, as the smallest and the largest eigenvalues for which the spectral equations admit a real solution: conditions (130) ensure that we find such points since they guaranty that we are approaching the edges from within the spectrum of $\widehat{\mathbf{U}}$ —otherwise the quantity ω_* would be undefined. By applying them to the real and imaginary parts of Eqs. (129), and setting $\operatorname{Re} g_\Psi(\lambda_*) = g_\Psi^*$ and $\operatorname{Re} g_\Omega(\lambda_*) = g_\Omega^*$, we obtain the following system of three algebraic equations

$$-\left(1-\frac{1}{\chi_p}\right)\frac{1}{(g_\Psi^*)^2}=-\frac{\omega_*}{\chi_p(g_\Psi^*)^2}+\frac{2g_\Omega^*}{\chi_p(g_\Psi^*)^3}-\frac{\chi_p}{\chi_m}\sum_c\frac{b_c h_c^2(h_c^2+\omega_*\gamma_c^2)}{\{1+\frac{\chi_p}{\chi_m}[\gamma_c^2 g_\Omega^*+h_c^2 g_\Psi^*]\}^2}$$

$$\frac{\omega_*}{\chi_p(g_\Omega^*)^2}=\frac{1}{\chi_p}\frac{1}{(g_\Psi^*)^2}+\frac{\chi_p}{\chi_m}\sum_c\frac{b_c\gamma_c^2(h_c^2+\omega_*\gamma_c^2)}{\{1+\frac{\chi_p}{\chi_m}[\gamma_c^2 g_\Omega^*+h_c^2 g_\Psi^*]\}^2} \tag{131}$$

$$\frac{1}{\chi_p g_\Omega^*}=\frac{\Delta}{e^{-2t}}\sum_c\frac{b_c}{\sigma_c^2}\gamma_c^2+\frac{1}{\chi_p}\frac{1}{g_\Psi^*}+\sum_c\frac{b_c\gamma_c^2}{1+\frac{\chi_p}{\chi_m}[\gamma_c^2 g_\Omega^*+h_c^2 g_\Psi^*]}.$$

The spectral edges are computed from the solutions of (131):

$$\boxed{\lambda_*=\sum_c b_c\varsigma_c+\sum_c\frac{b_c h_c^2}{1+\frac{\chi_p}{\chi_m}[\gamma_c^2 g_\Omega^*+h_c^2 g_\Psi^*]}-\left(1-\frac{1}{\chi_p}\right)\frac{1}{g_\Psi^*}-\frac{1}{\chi_p}\frac{g_\Omega^*}{(g_\Psi^*)^2}.} \tag{132}$$

Eqs. (131) and (132) are used to compute the curves in fig. 2**b** in the limit $t\to 0$, for the case of $C=2$ classes.

One can show that Eqs. (131) admit at most $2C+4$ solutions. In the following, we connect the solutions of (131) to the spectral edges $\lambda_{gen}, \lambda_{mem}^{(1)}, \lambda_{mem}^{(2)}$ defined in section 3.2. Similarly to [8], we take $\chi_p>\chi_m\gg 1$ and consider different scaling regimes for the resolvent function. We set $\chi_{pm}=\chi_p/\chi_m$.

First memorization bulk. We consider the scaling $g_\Psi=\mathcal{O}_{\chi_p}(1)$ and $g_\Omega=\mathcal{O}_{\chi_p}\left(\frac{1}{\chi_p}\right)$: let us set $\widetilde{g}_\Omega^*=\chi_p g_\Omega^*$. At leading order in $1/\chi_p$, the equations for the edges become

$$\begin{aligned}
&\frac{1}{(g_\Psi^*)^2}\approx\chi_{pm}\sum_c\frac{b_c h_c^4}{(1+\chi_{pm}h_c^2 g_\Psi^*)^2}\\
&\omega_*\approx\frac{1}{\chi_m}\sum_c\frac{b_c\gamma_c^2 h_c^2\widetilde{g}_\Omega^*}{(1+\chi_{pm}h_c^2 g_\Psi^*)^2}=\mathcal{O}_{\chi_m}\left(\frac{1}{\chi_m}\right)\\
&\frac{1}{\widetilde{g}_\Omega^*}\approx\frac{\Delta}{e^{-2t}}\sum_c\frac{b_c}{\sigma_c^2}\gamma_c^2+\sum_c\frac{b_c\gamma_c^2}{1+\chi_{pm}h_c^2 g_\Psi^*}\\
&\lambda_*\approx\sum_c b_c\varsigma_c+\sum_c\frac{b_c h_c^2}{1+\chi_{pm}h_c^2 g_\Psi^*}-\frac{1}{g_\Psi^*}-\frac{\widetilde{g}_\Omega^*}{(g_\Psi^*)^2}
\end{aligned} \tag{133}$$

In this limit, the equation for g_Ψ^* is decoupled from that for g_Ω^*. We observe that the equation for g_Ψ^* can be rearranged as an algebraic equation of order $2C$, thus yielding at most $2C$ values of spectral edges for C bulks. This regime identifies the t-independent memorization phase: the eigenvalue $\lambda_{mem}^{(1)}$ of section 3.2 is obtained from the solution g_Ψ^* that returns the largest spectral edge. This bulk has M eigenvalues, since it originates from the memorization term $\delta\mathbf{U}_{gep}$ in Eq. (57). Thus, it carries a weight χ_m/χ_p.

Note that from the first of (133) the scaling $g_\Psi^*\propto 1/\chi_{pm}$ emerges. Then, for $t\ll 1$ one finds that all the edges in this scaling regime satisfy $\lambda_*\propto\chi_{pm}$ and

$$\tau_{mem}^{(1)}=\frac{1}{2\lambda_{mem}^{(1)}}\propto\frac{\chi_m}{\chi_p} \tag{134}$$

as proven in [8] for the unimodal case.

Generalization bulk. Scaling $g_\Psi = \mathcal{O}_{\chi_p}(\frac{1}{\chi_p})$ and $g_\Omega = \mathcal{O}_{\chi_p}\left(\frac{1}{\chi_p}\right)$: we set $\widetilde{g}^*_\Psi = \chi_p g_\Psi$. We can find the spectral edges more easily from the spectral equations (129), which for this scaling read

$$\begin{aligned}\frac{\chi_p}{\widetilde{g}^*_\Psi} &= -\frac{\widetilde{g}^*_\Omega}{(\widetilde{g}^*_\Psi)^2} + (\varsigma - z) + \sum_c b_c h_c^2 \\ \frac{1}{\widetilde{g}^*_\Omega} &= \frac{\Delta}{e^{-2t}} \sum_c \frac{b_c}{\sigma_c^2}\gamma_c^2 + \frac{1}{\widetilde{g}^*_\Psi} + \sum_c b_c \gamma_c^2\end{aligned} \tag{135}$$

where we set $\varsigma = \sum_c b_c \varsigma_c$. Combining the two equations above, one finds the spectral equation of a Wishart matrix, and therefore the edges are those of a Marchenko–Pastur distribution [43]:

$$\lambda^{(g)}_\pm \approx \varsigma + \sum_c b_c h_c^2 + \left(\frac{\Delta}{e^{-2t}} \sum_c \frac{b_c}{\sigma_c^2}\gamma_c^2 + \sum_c b_c \gamma_c^2\right)(1 \pm \sqrt{\chi_p})^2 \tag{136}$$

These are the edges of the rightmost bulk shown in fig. 2**a**, which counts N eigenvalues and carries a weight $1/\chi_p$, since it is related to the population correlation $\widetilde{\mathbf{U}}_{gep}$ (Eq. (56) and see also Eq. (57)). It represents the generalization phase of the training: thus, we identify $\lambda_{gen} = \lambda^{(g)}_-$. We then have for $t \ll 1$

$$\tau_{gen} = \frac{1}{2\lambda_{gen}} \propto \frac{1}{\chi_p} \tag{137}$$

Combining (134), (137), we find the result of [8, 15, 39] for the generalization window

$$w_g \equiv \frac{\tau^{(1)}_{mem}}{\tau_{gen}} \propto \chi_m, \tag{138}$$

i.e., the generalization window grows linearly with the size of the dataset. Having verified the sample size scaling, in section 3.2 we study the dependence of the prefactor of this scaling law on the variances of the classes in a binary setting, to show how class diversity affects the generalization window.

Second memorization bulk. Scaling $g_\Psi = \mathcal{O}_{\chi_p}(\frac{\chi_m^2}{t\chi_p})$ and $g_\Omega = \mathcal{O}_{\chi_p}\left(\frac{\chi_m^2}{t\chi_p}\right)$: the choice of the scalings is motivated by fig. 11.

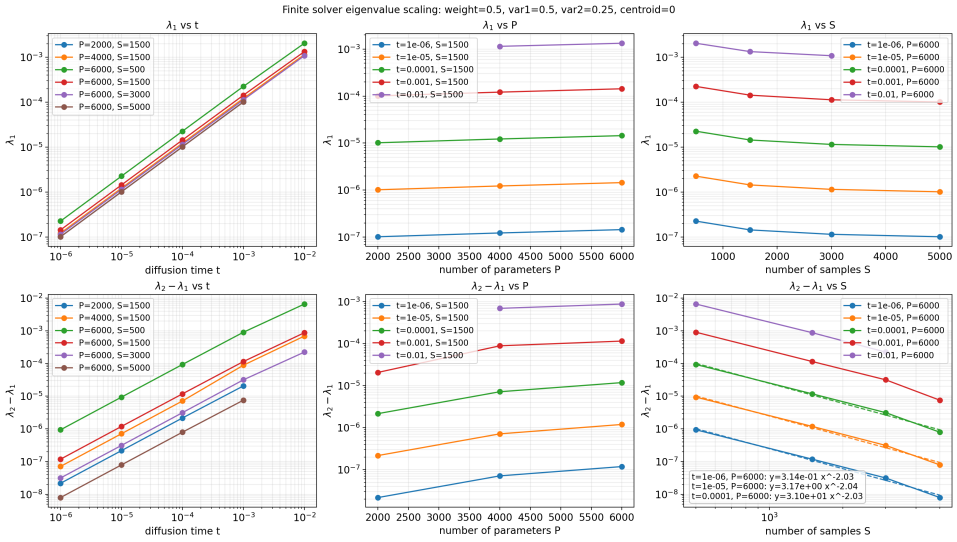


Figure 11: Check of the scaling of $\lambda^{(2)}_{mem}$.

We show that the eigenvalue $\lambda^{(2)}_{mem}$ scales linearly with t and converges to a constant value for a large number of hidden neurons and number of samples. We also show the width $\Delta\lambda$ of the leftmost

bulk in fig. 2**a** as a function of the same quantities: we find that it scales linearly with diffusion time and that it decays as the inverse square of the number of samples. We also find empirically that the weight of the leftmost bulk is $1/\chi_p$: then, considering our findings in fig. 11, we choose to scale the resolvent functions as above. We now proceed to expand the spectral equations (129) for $t \ll 1$ and $\chi_m \gg 1$: we get the following

$$\begin{aligned}\frac{t\chi_{pm}}{\chi_m \widetilde{g}^*_\Psi} &= -\frac{t\widetilde{g}^*_\Omega}{\chi_m^2(\widetilde{g}^*_\Psi)^2} + \varsigma - z + \frac{t}{\chi_m}\sum_c \frac{b_c h_c^2}{\gamma_c^2 \widetilde{g}^*_\Omega + h_c^2 \widetilde{g}^*_\Psi} \\ \frac{t}{\chi_m^2 \widetilde{g}^*_\Omega} &= \frac{\Delta}{e^{-2t}}\sum_c \frac{b_c}{\sigma_c^2}\gamma_c^2 + \frac{t}{\chi_m^2 \widetilde{g}^*_\Psi} + \frac{t}{\chi_m}\sum_c \frac{b_c \gamma_c^2}{\gamma_c^2 \widetilde{g}^*_\Omega + h_c^2 \widetilde{g}^*_\Psi}\end{aligned} \tag{139}$$

By summing these two equations together, we find for $t \ll 1$ and $\chi_m \gg 1$

$$\widetilde{g}^*_\Omega \approx \frac{z\widetilde{g}^*_\Psi}{2\sum_c b_c \frac{\gamma_c^2}{\sigma_c^2} t} \tag{140}$$

After some algebra, one finds that the edges are ($\widetilde{\varsigma} = t\,\varsigma$)

$$\lambda_\pm^{(m)} = \widetilde{\varsigma} t + \frac{\chi_{pm}\, t}{4\chi_m^3 \sum_c b_c \frac{\gamma_c^2}{\sigma_c^2}} \pm t\sqrt{\frac{\widetilde{\varsigma}\chi_{pm}}{2\chi_m^3 \sum_c b_c \frac{\gamma_c^2}{\sigma_c^2}} - \frac{2\sum_c b_c \frac{\gamma_c^2}{\sigma_c^2}}{\chi_m^3}\sum_c b_c \frac{h_c^2}{\gamma_c^2}} \tag{141}$$

In this last equation, we identify $\lambda^{(2)}_{mem} = \lambda^{(m)}_-$: we observe that it scales linearly in t. The bulk width $\lambda^{(m)}_+ - \lambda^{(m)}_-$ scales linearly with t and decays as $\chi_{pm}^{1/2}/\chi_m^{3/2} \propto \frac{1}{\chi_m^2}$.

We finally find that the scaling of the final memorization time is

$$\tau^{(2)}_{mem} = \frac{1}{2\lambda^{(2)}_{mem}} \propto \frac{1}{t} \tag{142}$$

and that the memorization window scales as

$$w_m = \frac{\tau^{(2)}_{mem}}{\tau^{(1)}_{mem}} \propto \frac{\chi_p}{\chi_m\, t}. \tag{143}$$

In section 3.2 we study the dependence of the prefactor of w_m on the variances of the classes and their sampling weights.

The remaining weight is $1 - \chi_m/\chi_p - 2/\chi_p$ and is carried by a delta peak: it represents eigenvectors that are orthogonal to all rows of the random features and GEP noise matrices. These modes do not affect training. The delta is located at $\lambda = \varsigma \equiv \sum_c b_c(\widetilde{\beta}_c - \beta_c - \frac{\Delta}{\sigma_c^2 e^{-2t}}\gamma_c^2) \propto t$.

I Theoretical curves of train and test errors

For the reader's convenience, we rewrite the equations of the train and test errors in Eqs. (3), (**??**):

$$\mathcal{E}_{train} = \frac{1}{NM}\sum_{\nu=1}^{M} \mathbb{E}_{\boldsymbol{\xi}}\left[\left\| \sqrt{\Delta_t}\mathbf{s}\left(\boldsymbol{x}_\nu e^{-t} + \sqrt{\Delta_t}\boldsymbol{\xi}\middle|\mathbf{A}, \mathbf{W}\right) + \boldsymbol{\xi}\right\|^2\right] \tag{144}$$

$$\mathcal{E}_{test} = \sum_c b_c\, \mathcal{E}^{(c)}_{test} \tag{145}$$

$$\mathcal{E}^{(c)}_{test} = \frac{1}{N}\mathbb{E}_{\boldsymbol{x}\sim\mathcal{N}(\mathbf{m}_c, \sigma_c^2\mathbf{I}_N),\,\boldsymbol{\xi}}\left[\| \sqrt{\Delta_t}\mathbf{s}\left(\boldsymbol{x}e^{-t} + \sqrt{\Delta_t}\boldsymbol{\xi}|\mathbf{A}, \mathbf{W}\right) + \boldsymbol{\xi}\|^2\right]$$

Training and test errors can be rewritten in the following form:

$$\mathcal{E}_{train} = 1 + \frac{\Delta_t}{N}\,\mathrm{Tr}(\mathbf{A}\mathbf{U}\mathbf{A}^\intercal) + \frac{2\sqrt{\Delta_t}}{N}\,\mathrm{Tr}(\mathbf{V}\mathbf{A}), \tag{146}$$

$$\mathcal{E}^{(c)}_{test} = 1 + \frac{\Delta_t}{N}\,\mathrm{Tr}(\mathbf{A}\widetilde{\mathbf{U}}_c\mathbf{A}^\intercal) + \frac{2\sqrt{\Delta_t}}{N}\,\mathrm{Tr}(\mathbf{V}_c\mathbf{A})$$

$$\simeq \mathcal{E}_{train} + \frac{\Delta_t}{N}\,\mathrm{Tr}(\mathbf{A}(\widetilde{\mathbf{U}}_c - \mathbf{U})\mathbf{A}^\intercal) + \frac{2\sqrt{\Delta_t}}{N}\,\mathrm{Tr}((\mathbf{V}_c - \mathbf{V})\mathbf{A}), \tag{147}$$

$$\mathcal{E}_{test} = \mathcal{E}_{train} + \frac{\Delta_t}{N}\,\mathrm{Tr}(\mathbf{A}(\widetilde{\mathbf{U}} - \mathbf{U})\mathbf{A}^\intercal).$$

By substituting into Eqs. (146), (147) the solution of the dynamics (7) and by expanding everything in the basis of eigenvectors of $\mathbf{U}$, we derived equations for the train and test errors:

$$\boxed{\begin{aligned}
&\mathcal{E}_{train}(\tau) = 1 - \frac{\Delta}{N}\sum_{q=1}^{P}\frac{1-e^{-4\lambda_q\tau}}{\lambda_q}\left\|\sum_{c=1}^{C}\frac{b_c}{\Gamma_c}\boldsymbol{\omega}_c^{(q)}\right\|^2,\\
&\mathcal{E}_{test}(\tau) = \mathcal{E}_{train}(\tau) + \frac{\Delta}{N}\sum_{q,p}\frac{\left(1-e^{-2\lambda_q\tau}\right)\left(1-e^{-2\lambda_p\tau}\right)}{\lambda_q\lambda_p}\\
&\times\left(\sum_c b_c\frac{\boldsymbol{\omega}_c^{(q)}}{\Gamma_c}\right)^\intercal\left(\sum_c b_c\frac{\boldsymbol{\omega}_c^{(p)}}{\Gamma_c}\right)\left(\boldsymbol{\psi}_q^\intercal\widetilde{\mathbf{U}}\boldsymbol{\psi}_p - \lambda_p\delta_{qp}\right),\\
&\mathcal{E}^{(c)}_{test}(\tau) = \mathcal{E}_{train}(\tau) + \frac{\Delta}{N}\sum_{q,p}\frac{\left(1-e^{-2\lambda_q\tau}\right)\left(1-e^{-2\lambda_p\tau}\right)}{\lambda_q\lambda_p}\\
&\times\left(\sum_c b_c\frac{\boldsymbol{\omega}_c^{(q)}}{\Gamma_c}\right)^\intercal\left(\sum_c b_c\frac{\boldsymbol{\omega}_c^{(p)}}{\Gamma_c}\right)\left(\boldsymbol{\psi}_q^\intercal\widetilde{\mathbf{U}}_c\boldsymbol{\psi}_p - \lambda_p\delta_{qp}\right)\\
&-\frac{2\Delta}{N}\sum_q\frac{1-e^{-2\,\lambda_q\tau}}{\lambda_q}\left(\sum_{c'}b_{c'}\frac{\boldsymbol{\omega}_{c'}^{(q)}}{\Gamma_{c'}}\right)^\intercal\left(\frac{\boldsymbol{\omega}_c^{(q)}}{\Gamma_c} - \left(\sum_{c'}b_{c'}\frac{\boldsymbol{\omega}_{c'}^{(q)}}{\Gamma_{c'}}\right)\right),
\end{aligned}} \tag{148}$$

where $\{\lambda_q, \boldsymbol{\psi}_q\}$ are eigenvalues and eigenvectors of $\mathbf{U}$, and we introduced $\boldsymbol{\omega}_c^{(q)} = \frac{1}{\sqrt{N}}\mathbf{W}^\intercal(\boldsymbol{\psi}_q \odot \widetilde{\boldsymbol{\gamma}}_c)$. The formulas in Eqs. (148), with the feature correlation matrices replaced by their GEP expressions, are our *semi-analytical* curves for the train and test errors. We name them so because we need to numerically diagonalize the matrix $\mathbf{U}_{gep}$ to use them. The curves for the test errors were used in sections 3.2 and 3.2.1 to represent our theoretical predictions.

For test errors, in general we cannot write asymptotic expressions in terms of the solution of the saddle point equations (122); we can only do that for the train error. We derived the following asymptotic formula for the train error, in the joint limits $N, P, M \to \infty$ with fixed ratios $\chi_p = P/N$ and $\chi_m = M/N$:

$$\boxed{\mathcal{E}_{train}(\tau) = 1 - \frac{\Delta\chi_p}{e^{-2t}}\sum_{cc'}\frac{b_c b_{c'}}{\sigma_c\sigma_{c'}}\int\frac{d\lambda}{\lambda}\left(1-e^{-4\lambda\tau}\right)\rho^{(\Omega)}_{cc'}(\lambda)} \tag{149}$$

where the spectral densities $\rho^{(\Omega)}_{cc'}$ are computed from the resolvent functions $g^{(\Omega)}_{cc'}$ in Eq. (122):

$$\rho^{(\Omega)}_{cc'}(\lambda) = \frac{1}{\pi} \lim_{\epsilon \to 0_+} \operatorname{Im} g^{(\Omega)}_{cc'}(\lambda + i\epsilon)$$

We refer to Eq. (149) as our *fully analytical* formula for the train error. We derived (149) from the first equation in (148), noting that the dot products $\omega_c(\lambda) \cdot \omega_{c'}(\lambda)$ featuring in (148) can be computed from the imaginary part of $g^{cc'}_{\Omega}$. Indeed, since by definition one has

$$\begin{aligned} g^{cc'}_{\Omega}(z) &= \frac{1}{N^2} \sum_{q=1}^{P} \frac{\boldsymbol{\psi}(\lambda_q)^{\intercal} (\mathbf{W}^{\intercal} \odot \mathbf{1}_N \boldsymbol{\gamma}_{c'})(\boldsymbol{\gamma}_c \mathbf{1}_N^{\intercal} \odot \mathbf{W}) \boldsymbol{\psi}(\lambda_q)}{\lambda_q - z} \\ &\equiv \frac{\sigma_c \sigma_{c'} e^{-2t}}{\Gamma_c \Gamma_{c'}} \int d\lambda \frac{\rho(\lambda)\, \boldsymbol{\omega}_c(\lambda) \cdot \boldsymbol{\omega}_{c'}(\lambda)}{\lambda - z} \end{aligned}$$

then it follows that

$$\boldsymbol{\omega}_c(\lambda) \cdot \boldsymbol{\omega}_{c'}(\lambda) = \frac{\Gamma_c \Gamma_{c'}}{\sigma_c \sigma_{c'} e^{-2t}} \frac{\rho^{cc'}_{\Omega}(\lambda)}{\rho(\lambda)}. \tag{150}$$

We indicate with $\boldsymbol{\omega}(\lambda)$ the continuous limit of the $\boldsymbol{\omega}_q$ defined above. In the limit of subextensive centroids, $\|\boldsymbol{m}_c\| = \mathcal{O}_N(1)$, the vector Hermite coefficients become homogeneous and thus (150) turns into

$$\boldsymbol{\omega}_c(\lambda) \cdot \boldsymbol{\omega}_{c'}(\lambda) = \frac{\Gamma_c \Gamma_{c'} \gamma_c \gamma_{c'}}{\sigma_c \sigma_{c'} e^{-2t}} \frac{\rho_{\Omega}(\lambda)}{\rho(\lambda)} \tag{151}$$

and the analytical train error formula simplifies to

$$\mathcal{E}_{train}(\tau) = 1 - \frac{\Delta \chi_p}{e^{-2t}} \left(\sum_{c=1}^{C} b_c \frac{\gamma_c}{\sigma_c} \right)^2 \int \frac{d\lambda}{\lambda} \left(1 - e^{-4\lambda\tau}\right) \rho_{\Omega}(\lambda). \tag{152}$$

J Score MSE

We write the formula for the MSE of the score:

$$\mathcal{E}_{score} = \sum_c b_c \mathcal{E}^{(c)}_{score}, \qquad \mathcal{E}^{(c)}_{score} = \frac{1}{N} \mathbb{E}_{\boldsymbol{x} \sim \mathcal{N}_c(t)} \left[\left\| \mathbf{A}\, \phi\left(\frac{1}{\sqrt{N}} \mathbf{W} \boldsymbol{x} \right) - \boldsymbol{s}_{true}(\boldsymbol{x}, t) \right\|^2 \right] \tag{153}$$

The MSE on the score is a proxy for the quality of the generative model: indeed, the Kullback-Leibler (KL) divergence between the prior distribution $\mathcal{P}^*$ and the sampling distribution learned by the diffusion model $\widetilde{\mathcal{P}}$ is upper-bounded by the integral over diffusion time of the score error [52]:

$$D_{KL}(\widetilde{\mathcal{P}} | \mathcal{P}^*) \equiv - \int d\boldsymbol{x} \mathcal{P}^*(\boldsymbol{x}) \log\Big(\widetilde{\mathcal{P}}(\boldsymbol{x}) / \mathcal{P}^*(\boldsymbol{x}) \Big) \leq \frac{N}{2} \int dt \mathcal{E}_{score}(t) \tag{154}$$

Given the hypothesis of well-separated classes, the condition in (154) holds also for the single components of the mixture:

$$D_{KL}(\widetilde{\mathcal{P}}_c | \mathcal{P}^*_c) \equiv - \int d\boldsymbol{x} \mathcal{P}^*_c(\boldsymbol{x}) \log\Big(\widetilde{\mathcal{P}}_c(\boldsymbol{x}) / \mathcal{P}^*_c(\boldsymbol{x}) \Big) \leq \frac{N}{2} \int dt \mathcal{E}^{(c)}_{score}(t). \tag{155}$$

We can compute semi-analytical formulas analogous to those in (148) for the score MSEs. The score of a Gaussian mixture such as that in (6) has the following expression:

$$\begin{aligned} &\boldsymbol{s}_{true}(\boldsymbol{x}, t) = \sum_{c=1}^{C} b_c \boldsymbol{s}^{(c)}_{true}(\boldsymbol{x}, t) \qquad \boldsymbol{s}^{(c)}_{true}(\boldsymbol{x}, t) = - \frac{\boldsymbol{x} - e^{-t} \boldsymbol{m}_c}{\Gamma^2_c(t)}\, \zeta_c(\boldsymbol{x}, t) \\ &\zeta_c(\boldsymbol{x}, t) = \frac{\mathcal{P}_c(\boldsymbol{x}, t)}{\sum_{c'=1}^{C} b_{c'} \mathcal{P}_{c'}(\boldsymbol{x}, t)}, \qquad \mathcal{P}_c(\boldsymbol{x}, t) = \frac{e^{-\frac{\left\| \boldsymbol{x} - e^{-t} \boldsymbol{m}_c \right\|^2}{2\Gamma^2_c(t)}}}{(2\pi \Gamma^2_c(t))^{N/2}} \end{aligned} \tag{156}$$

For generic mixtures, the components of the true score interfere with each other through the functions ζ_c. In the case of well-separated mixtures, one gets $\zeta_c = 1/b_c$ for any datapoint sampled from that class: then, the scores are the scores that the components of the mixture would have in isolation,

$$b_c \mathbf{s}^{(c)}_{true}(\boldsymbol{x}, t) \approx -\frac{\boldsymbol{x} - \mathbf{m}_c e^{-t}}{\Gamma^2_c(t)} \equiv \boldsymbol{s}_c(\boldsymbol{x}, t) \qquad \boldsymbol{x} \text{ belonging to class } c. \tag{157}$$

We can define the effective domain for classes by looking at Eqs. (156): all exponential terms should be vanishing for classes $c' \neq c$ compared to class c, and this is achieved for values of $\boldsymbol{x}$ such that

$$\frac{1}{\Gamma^2_{c'}(t) N} \|\boldsymbol{x} - \mathbf{m}_{c'} e^{-t}\|^2 - \frac{1}{\Gamma^2_c(t) N} \|\boldsymbol{x} - \mathbf{m}_c e^{-t}\|^2 + \log(\Gamma_{c'}/\Gamma_c) > 0, \qquad \forall c' \neq c, N \to \infty \tag{158}$$

The boundaries of the effective domains are regions such that

$$\left| \frac{1}{\Gamma^2_{c'}(t)} \|\boldsymbol{x} - \mathbf{m}_{c'} e^{-t}\|^2 - \frac{1}{\Gamma^2_c(t)} \|\boldsymbol{x} - \mathbf{m}_c e^{-t}\|^2 + N \log(\Gamma_{c'}/\Gamma_c) \right| = \mathcal{O}_N(1) \qquad N \to \infty \tag{159}$$

In the high-dimensional limit $N \gg 1$, samples $\boldsymbol{x}$ belong with high probability to the bulks of the effective domains, the boundaries are atypical. We consider the case of well-separated mixtures, where the inequalities (158) hold for all classes: then the score MSE in (153) reduces to

$$\mathcal{E}_{score} = \sum_c b_c \mathcal{E}^{(c)}_{score}, \qquad \mathcal{E}^{(c)}_{score} \simeq \frac{1}{N} \mathbb{E}_{\boldsymbol{x} \sim \mathcal{N}_c(t)} \left[\left\| \mathbf{A}\, \phi\left(\frac{1}{\sqrt{N}} \mathbf{W} \boldsymbol{x} \right) + \frac{\boldsymbol{x} - e^{-t} \boldsymbol{m}_c}{\Gamma^2_c(t)} \right\|^2 \right] \tag{160}$$

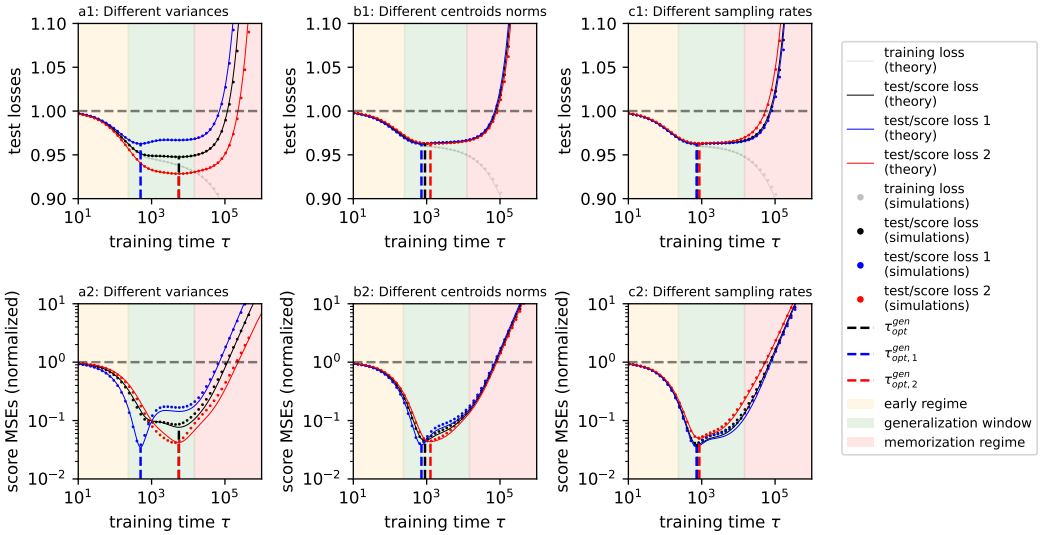


Figure 12: Comparison between the test losses and the score MSEs. Parameters as in figure 3.

Finally, by following the steps of Appendix I, we obtain

$$\begin{aligned} \mathcal{E}^{(c)}_{score}(\tau) &= \frac{1}{\Gamma^2_c(t)} - \frac{2}{\Gamma_c N} \sum_q \frac{1 - e^{-2\lambda_q \tau}}{\lambda_q} \left(\sum_{c''} \frac{b_{c''}}{\Gamma_{c''}} \boldsymbol{\omega}^q_{c''} \right)^{\intercal} \boldsymbol{\omega}^q_c \\ &+ \frac{1}{N} \sum_{pq} \frac{(1 - e^{-2\lambda_p \tau})(1 - e^{-2\lambda_q \tau})}{\lambda_p \lambda_q} \left(\sum_{c''} \frac{b_{c''}}{\Gamma_{c''}} \boldsymbol{\omega}^p_{c''} \cdot \sum_{c'''} \frac{b_{c'''}}{\Gamma_{c'''}} \boldsymbol{\omega}^q_{c'''} \right) \boldsymbol{\psi}_p^{\intercal} \widetilde{\mathbf{U}}_c \boldsymbol{\psi}_q \\ &\equiv \frac{1}{\Gamma^2_c(t)} + \frac{1}{\Delta_t} (\mathcal{E}^{(c)}_{test}(\tau) - 1) \end{aligned} \tag{161}$$

We verified that in the limit of well-separated classes, the score MSE of any class c is just the test loss of that class plus a diffusion time and a class-dependent constant shift, along with a rescaling multiplicative factor dependent on diffusion time, in agreement with Tweedie's identity. This implies that the class-specific generalization times and memorization times defined in (16) for the test losses are the same for the score MSEs. In figure 12 we compare the test loss shown in figure 3 with the score MSE.

Appendices for experiments

K Numerical experiments on Random Features

K.1 Training of RF model

Training of the RF model was conducted using gradient descent: for $k = 1, \dots, N_{epoch}$, we used

$$\mathbf{A}_{k+1} = \mathbf{A}_k - \eta \frac{\partial \mathcal{E}_{train}}{\partial \mathbf{A}}\bigg|_{\mathbf{A}=\mathbf{A}_k}$$

with $\mathcal{E}_{train}$ given by Eq. (3). In all experiments, we set $\eta = 5 \times 10^{-5} \frac{N}{\Delta_t}$, where N is data dimension, and $\Delta_t = 1 - e^{-2t}$ (compare with Eq. (26) to understand the origin of this scaling choice of the learning rate). We set $N_{epoch} = 2 \times 10^6$ and 600 log-spaced sampling times. We fix $P = 6000$, $N = 100$ and $M = 3000$ for the parameters in sections 3.2, 3.2.1.

For each choice of data distribution parameters, we conducted several runs with different instantiations of the RF model and the dataset, namely different draws of the matrix $\mathbf{W}$ and the dataset $\mathbf{X}$, and then obtained curves for train, test and score errors versus training time by averaging over different runs. We applied a Gaussian smoothing to each error curve of each run, to smooth out fluctuations at small scales due to GD discrete steps. Generalization and memorization times were computed using (16) on the curves averaged over different runs. As the dynamics converge to a unique readout matrix, asymptotically independent of the initial condition (see Eq. (26)), we did not average over different initial conditions.

An experiment with 5 runs, with the parameters specified above, requires approximately four hours and a half on an $A40$ GPU.

K.2 Numerical diagonalization of $\mathbf{U}_{gep}, \widetilde{\mathbf{U}}_{gep}$

To evaluate the semi-analytical curves in Eqs. (148),(161), for each choice of Gaussian mixture, we obtained the eigenvalues and eigenvectors of $N_{diag} = 10$ instances of the GEP matrices $\mathbf{U}_{gep}$ and $\widetilde{\mathbf{U}}_{gep}$. In sections 3.2, 3.2.1, we set $P = 6000$ for the number of hidden neurons, $N = 100$ for data dimension and $M = 3000$ for the number of samples. In section 4, we set $P = 10000$, $N = 100$, $M = 500$.

The numerical diagonalization of instances of $\mathbf{U}_{gep}, \widetilde{\mathbf{U}}_{gep}$ requires roughly 10 minutes on a state-of-the-art CPU.

K.3 Solution of the spectral equations

Algorithm 1 Solver of Eqs. (129)

Require: Hermite coefficients $\{(\beta_k, \tilde{\beta}_k, \gamma_k)\}_{k=1}^{C}$, diagonal terms $d_k = \tilde{\beta}_k - \beta_k - \frac{\Delta}{2e^{-2\tau}}\gamma_k^2$, grid $\{\lambda_j\}_{j=1}^{N_{\text{values}}}$, continuation schedule $\mathcal{E} = \text{Concat}(\text{Log-Spaced}([100, 10^{-9}], N_{eps}), \{0\}]$.
Ensure: real parts $\{\text{Re}\, g_\Psi(\lambda_j),\ \text{Re}\, g_\Omega(\lambda_j)\}$ and spectral densities $\{\rho(\lambda_j),\ \rho_\Omega(\lambda_j)\}$

for $j = 1$ **to** N_{values} **do** ▷ sweep over spectral grid
  $\lambda \leftarrow \lambda_j$
  $z \leftarrow \lambda + i\,\varepsilon_0$ ▷ initialize at largest imaginary shift
  $g_\Psi^{(0)} \leftarrow -1/z, \quad g_\Omega^{(0)} \leftarrow -1/z$ ▷ free-theory initial guess
  for $\varepsilon \in \mathcal{E}$ **do** ▷ decrease imaginary part to 0
    $z \leftarrow \lambda + i\,\varepsilon$
    $(g_\Psi, g_\Omega) \leftarrow \textsc{Solve}\Big(F_\Psi(\cdot,\cdot;\, z) = 0,\ F_\Omega(\cdot,\cdot;\, z) = 0,\ \text{init} = (g_\Psi^{(0)}, g_\Omega^{(0)})\Big)$ ▷ Levenberg–Marquardt
    $g_\Psi^{(0)} \leftarrow g_\Psi, \quad g_\Omega^{(0)} \leftarrow g_\Omega$ ▷ warm start for next ε
  end for
  $\text{Re}\, g_\Psi(\lambda_j) \leftarrow \text{Re}(g_\Psi)$
  $\rho(\lambda_j) \leftarrow \frac{1}{\pi}\,\text{Im}(g_\Psi)$
  $\text{Re}\, g_\Omega(\lambda_j) \leftarrow \text{Re}(g_\Omega)$
  $\rho_\Omega(\lambda_j) \leftarrow \frac{1}{\pi}\,\text{Im}(g_\Omega)$
end for

To solve Eqs. (129), we used the following algorithm: for each of 10^4 log-spaced values of $\lambda \in [10^{-8}, 10^2]$, we took $z = \lambda + i\varepsilon$ and iteratively solved (132) for decreasing values of ε until $\varepsilon = 0$, starting from $\varepsilon_0 = 100$ with initial conditions $g_{start}(z) = -1/z$ and $g_\Omega^{(start)} = -1/z$. The spectral densities $\rho(\lambda), \rho_\Omega(\lambda)$ are evaluated with formula (80). Spectral density values below 10^{-6} are truncated to zero.

The choice of $\varepsilon_0 \gg 1$ ensures the selection of the correct branch of $g(z)$ (in random matrix theory, the resolvent function is a multivalued function featuring different branches, see [43]), uniquely identified by $g(z) \approx -1/z$ for $|z| \gg 1$.

To solve Eqs. (131) to compute the edges through (132), we computed the initial conditions from the solution of (132), by selecting values of λ in the grid where $\rho(\lambda)$ acquires a nonzero value.

In algorithm 1 we report pseudocode for the solver algorithm. The solver takes around 3 minutes on a state-of-the-art CPU to solve the equations on a grid of $N_{values} = 10^4$ values of λ.

L Numerical experiments on Fashion MNIST

L.1 Additional simulations framework details

Model. A standard DDPM [22] was employed on the FashionMNIST dataset [65], with a U-Net [46] serving as the backbone. The network follows a U-shaped encoder–decoder structure connected by skip connections across four resolution levels (64, 128, 256, 256 channels, respectively). Each level contains ResNet blocks [21] with a dropout rate of 0.1 [54], conditioned on the diffusion time step through sinusoidal time embeddings [22, 59] injected additively into each block. The encoder is composed of two such blocks per level, and the decoder of three. Four-head self-attention with GroupNorm pre-normalization is applied within the ResNet blocks at the two intermediate resolutions. At the deepest level, a two-block bottleneck bridges the encoder and the decoder, with self-attention enabled in the first block and disabled in the second.

Training. Each model is trained on a binary subset of FashionMNIST: the Sneaker class, that exhibits the lowest pixel-wise variance, is paired with each of the other classes, yielding nine class pairs in total. Each pair is trained at three Sneaker proportions (25%, 50%, 75%) on 8,000 images on a single core of an NVIDIA A100 GPU for approximately one week, reaching approximately

4×10^6 gradient steps. In this study, a linear noise schedule is adopted, with the parameters ranging from $\beta_1 = 10^{-4}$ to $\beta_T = 2 \times 10^{-2}$, and $T = 1000$ discrete steps. The Adam optimizer ($\eta = 10^{-4}$) is utilized to minimize the mean squared error (MSE) between the predicted and the true noise [28]. 32 model checkpoints are saved at log-spaced intervals (5 per scale) per run to capture both early and late training dynamics.

Generation. 500 samples per pair were generated with the DDPM ancestral sampler [22]. All checkpoints in one to two hours with a single NVIDIA A100 or A40 GPU are included. Starting from $\boldsymbol{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_N)$, denoising through all T reverse steps was iteratively performed by the model. At each step $t \in \{T, \dots, 1\}$, the update is

$$\boldsymbol{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\boldsymbol{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\varepsilon}_\theta(\boldsymbol{x}_t, t) \right) + \sqrt{\beta_t}\, \boldsymbol{z}_t,$$

where $\boldsymbol{z}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_N)$ for $t > 1$ and $z_t = 0$ for $t = 1$, $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s = \prod_{s=1}^{t} (1 - \beta_s)$ is the cumulative product, and $\boldsymbol{\varepsilon}_\theta(\boldsymbol{x}_t, t)$ is the U-Net's predicted noise. Examples of generated samples can be found in Appendix L.2.

Evaluation. In accordance with the nearest-neighbor criterion outlined in [8, 20, 66], a generated sample $\boldsymbol{x}_c$ from class c is deemed memorized if

$$r(\boldsymbol{x}_c) = \frac{\|\boldsymbol{x}_c - \mathrm{nn}_1(\boldsymbol{x}_c)\|_2}{\|\boldsymbol{x}_c - \mathrm{nn}_2(\boldsymbol{x}_c)\|_2} < \frac{1}{3},$$

where $\mathrm{nn}_i(\boldsymbol{x}_c)$ denotes the i-th nearest neighbor of $\boldsymbol{x}_c$ in the training set under the ℓ_2 norm. The per-class memorization fraction is defined as the expected memorization rate under the estimated data distribution $p_\theta^c(t)$ at time t,

$$f_{\mathrm{mem}}^c(t) = \mathbb{E}_{\boldsymbol{x}_c \sim p_\theta^c(t)} \left[\mathbb{I}\big(r(\boldsymbol{x}_c) < \tfrac{1}{3}\big) \right],$$

and is empirically estimated by its mean over n_c generated samples. We define the threshold-crossing time $t_{\mathrm{mem}}^c = \inf\{t : f_{\mathrm{mem}}^c(t) \geq 1/3\}$, so the Sneaker-partner memorization gap

$$\Delta t_{\mathrm{mem}}(c_p) = t_{\mathrm{mem}}^{\mathrm{snk}} - t_{\mathrm{mem}}^{c_p} \tag{162}$$

is tracked across all log-spaced checkpoints and class ratios.

To assign each generated sample to its class, a dedicated binary ResNet-18 classifier [21] is trained for each of the nine Sneaker–partner pairs. The first convolutional layer is replaced with a single-channel counterpart to accommodate Fashion MNIST's grayscale input, and the network is trained with the same input preprocessing as the diffusion model (class-pair mean subtracted, unit standard deviation). Training uses AdamW [35] with cosine annealing [34] over 15 epochs, differential learning rates (backbone: 10^{-5}, head: 10^{-4}) [13, 23], weight decay 0.01 [30], and label smoothing 0.1 [56]. The classifiers used for class assignment are trained on a 90/10 train–validation split; using an 80/20 split led to no material difference. Across all class proportions, all nine classifiers achieved above 96.0% macro F1 and 97.3% balanced accuracy. The reported lowest values were observed for the visually similar Sneaker–Ankle Boot pair, confirming reliable class attribution of generated samples under the primary imbalance conditions examined.

L.2 Supplementary figures

Generated samples and memorization dynamics. We first visualize a representative Sneaker–Bag pair in Fig. 13 to illustrate the transition from generalization to memorization. The figure shows generated samples at selected checkpoints alongside their first and second nearest training examples, providing an intuitive view of the sample-level memorization criterion defined in section 4. Under this criterion, the memorized sample is highlighted in red. The bottom panel traces the corresponding nearest-neighbor dynamics: solid and dashed curves denote the mean distances to the first and second nearest neighbors, respectively, while vertical dotted lines mark the class-specific memorization times. Figs. 14**a** to 14**h** report the same analysis for the remaining eight Fashion MNIST class pairs.

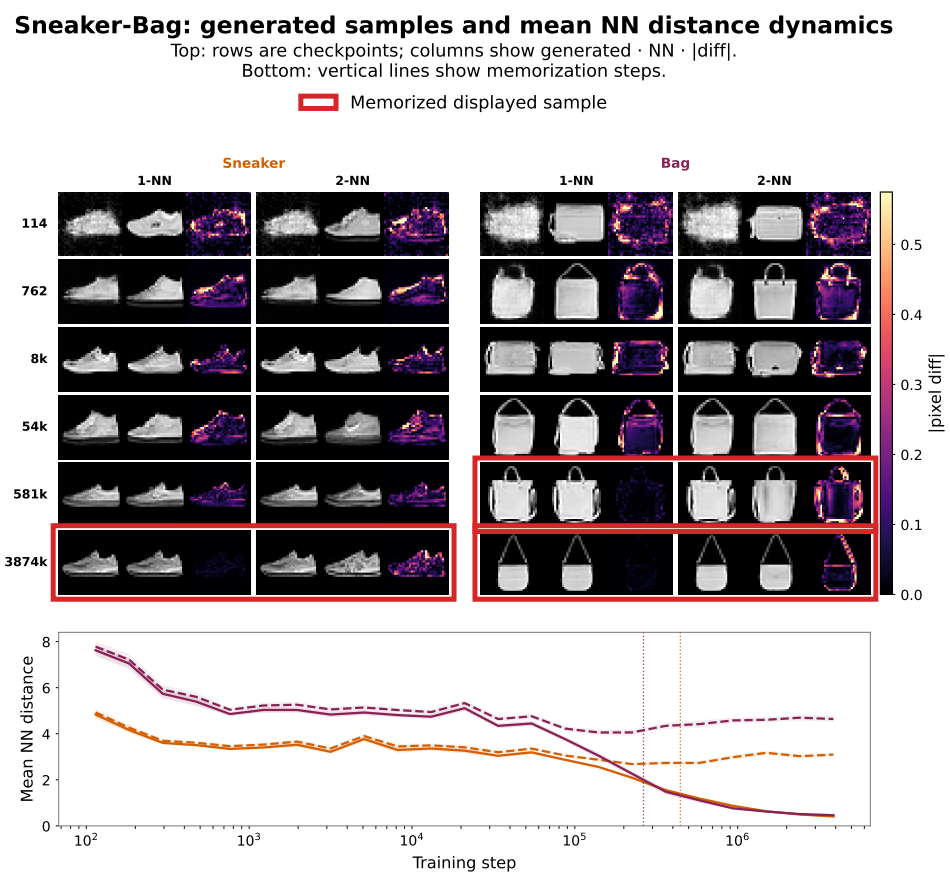
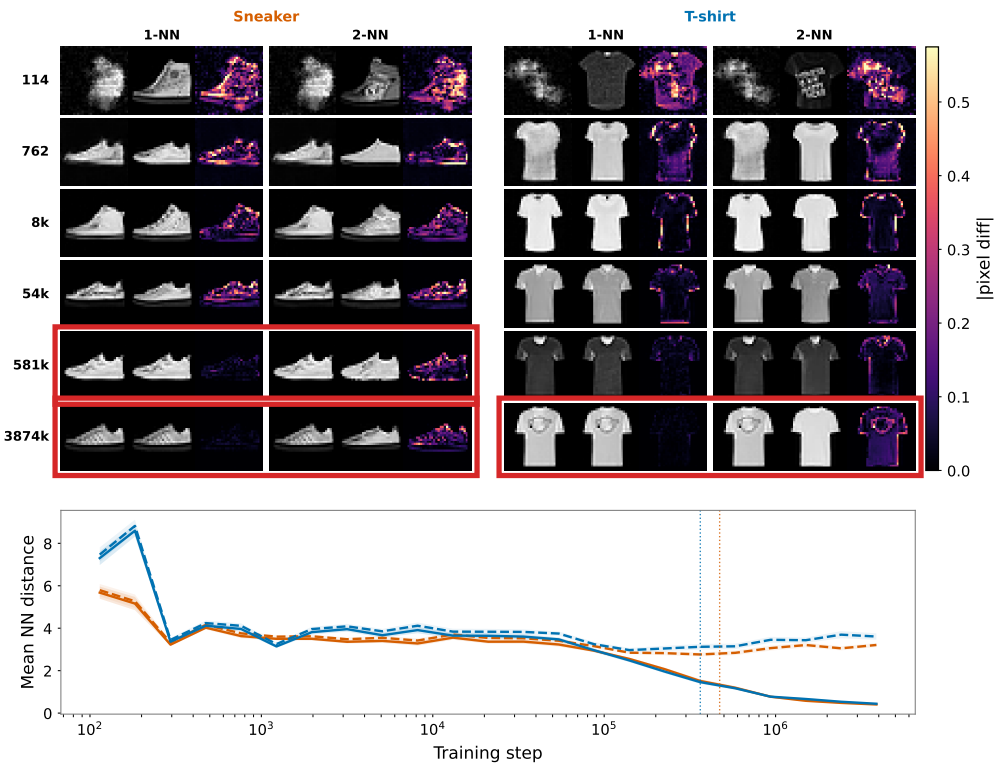
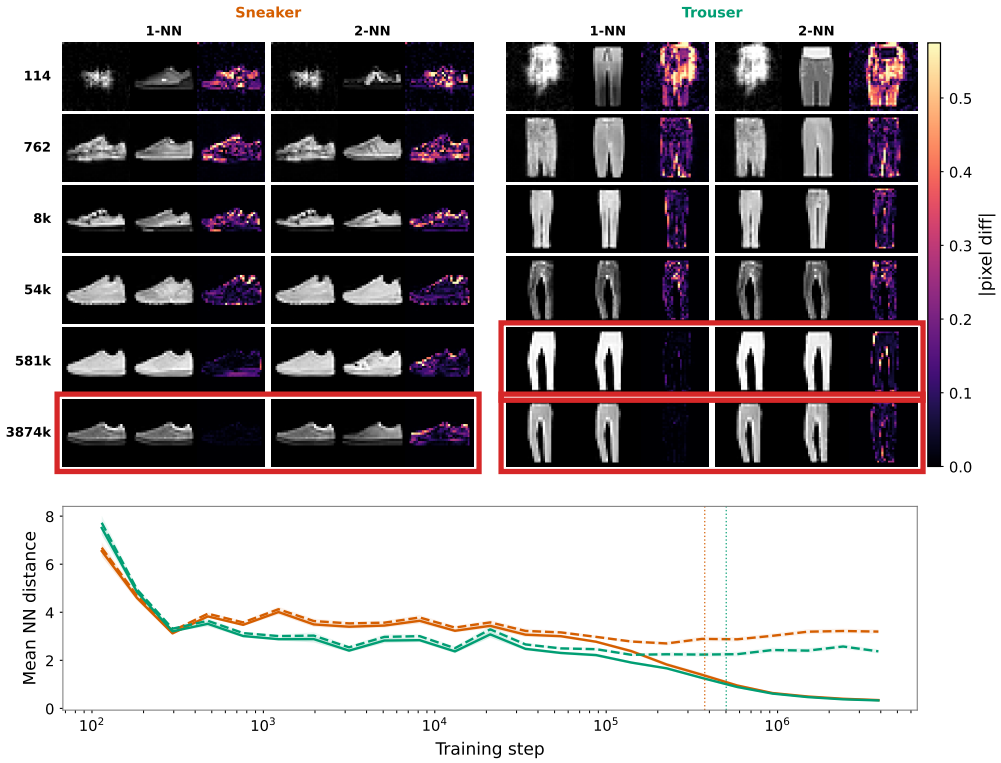


Figure 13: Overview examples for the Sneaker and Bag classes.



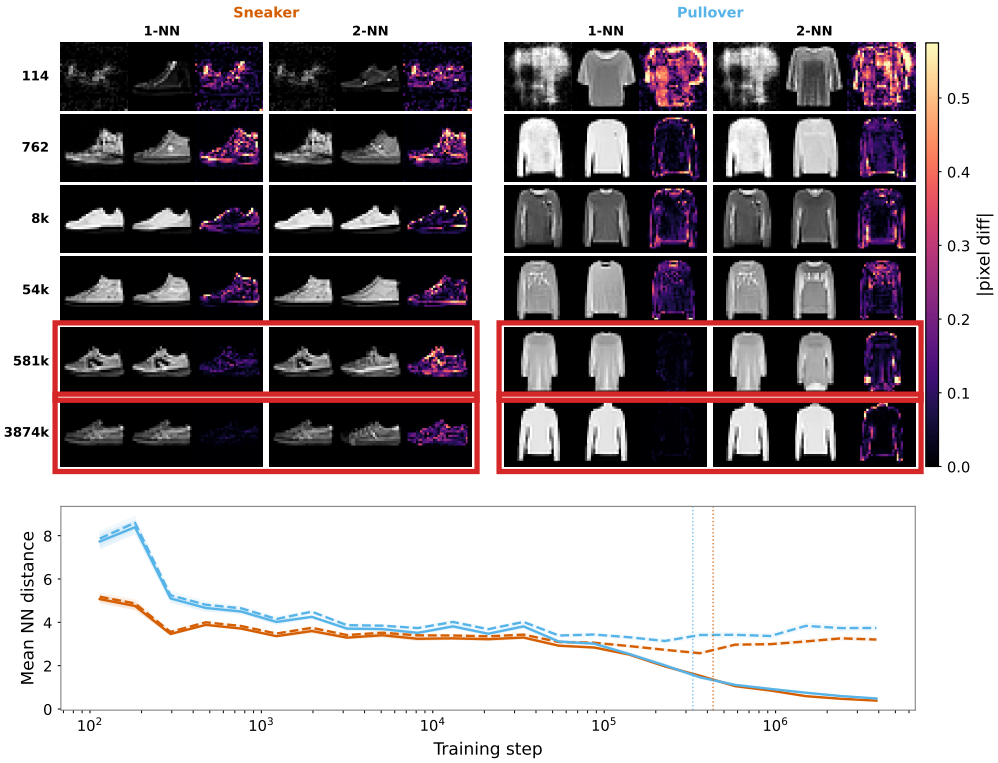
a T-shirt/top

Figure 14: Overview examples for Fashion MNIST classes (Part 1).



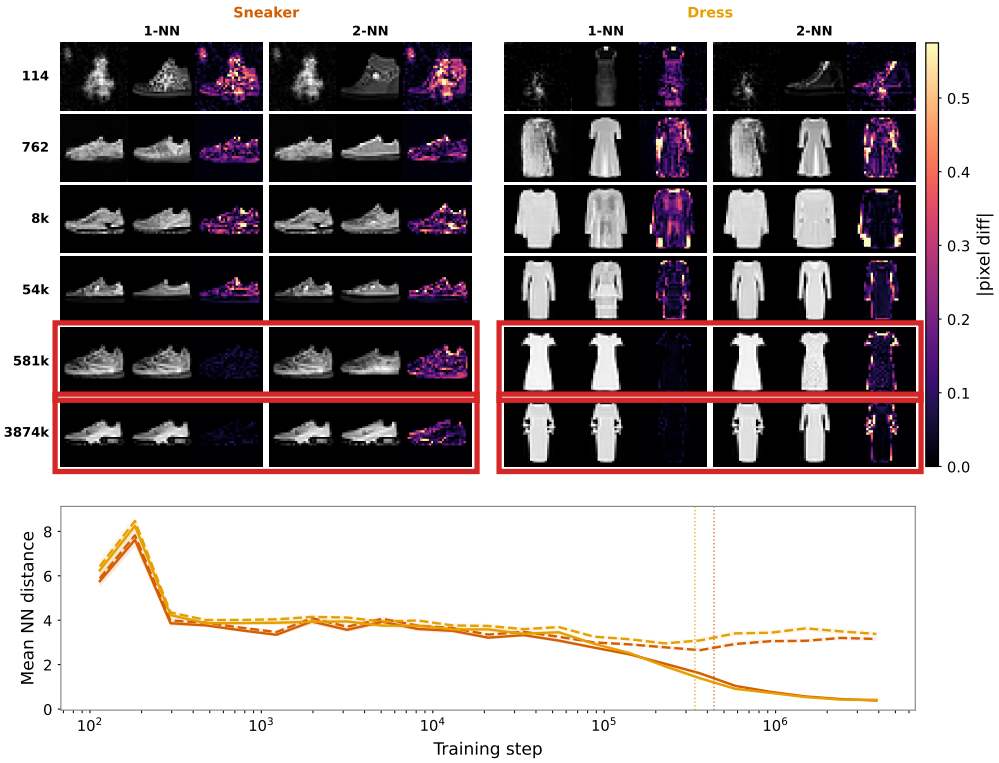
b Trouser

Figure 14: Overview examples for Fashion MNIST classes (Part 2).



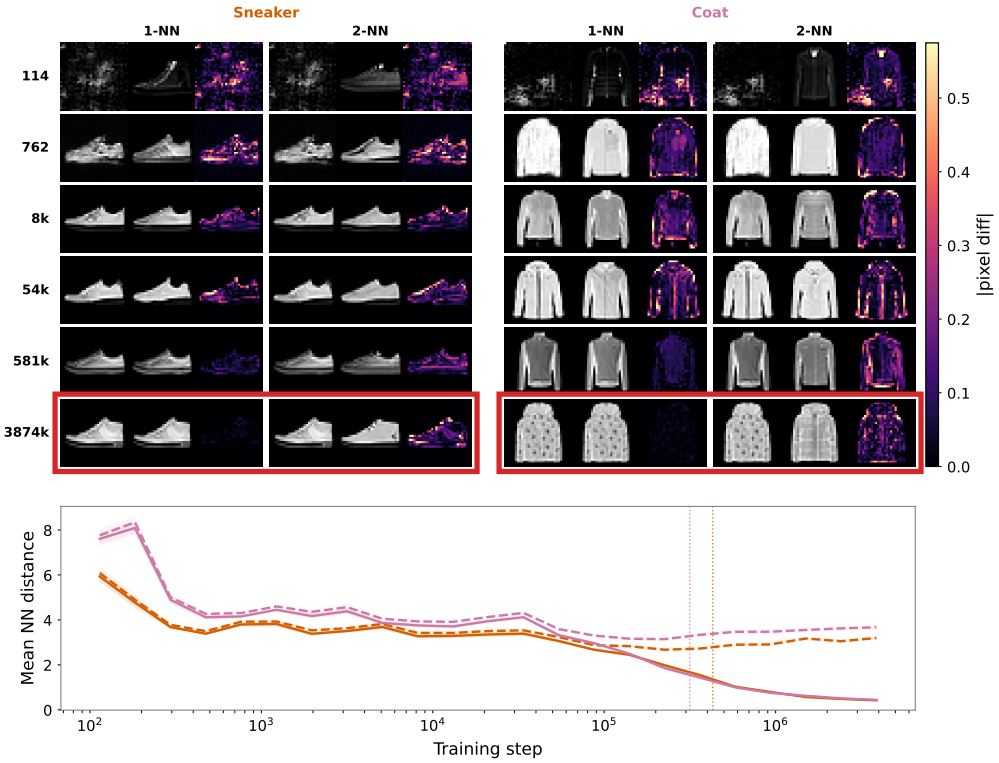
c Pullover

Figure 14: Overview examples for Fashion MNIST classes (Part 3).



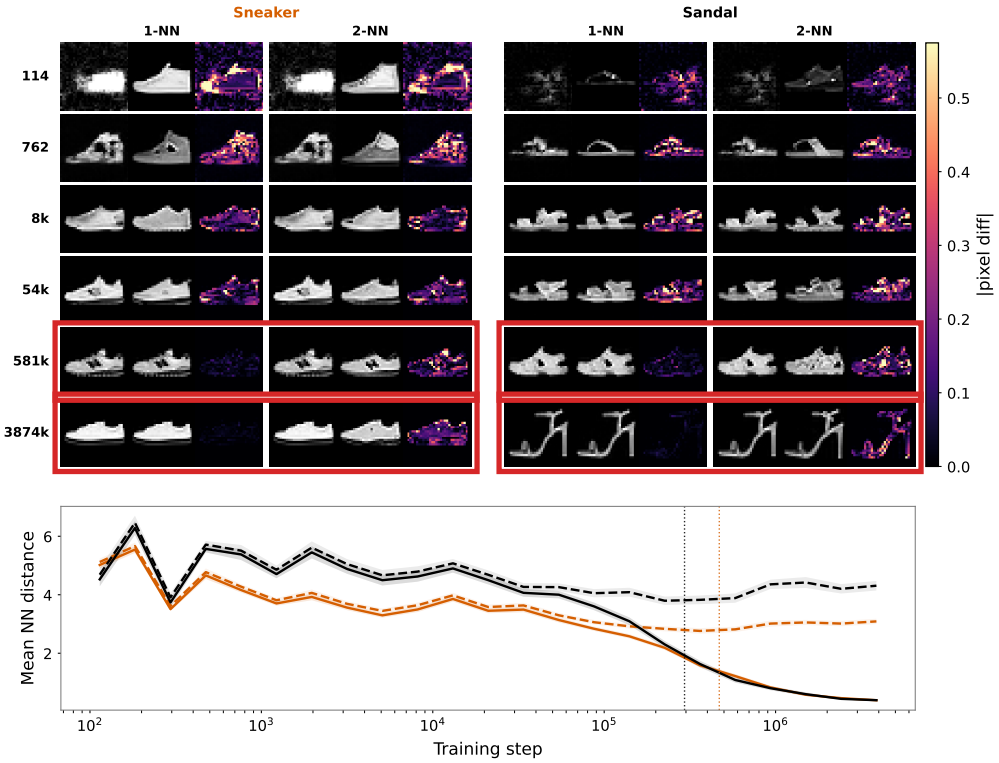
d Dress

Figure 14: Overview examples for Fashion MNIST classes (Part 4).



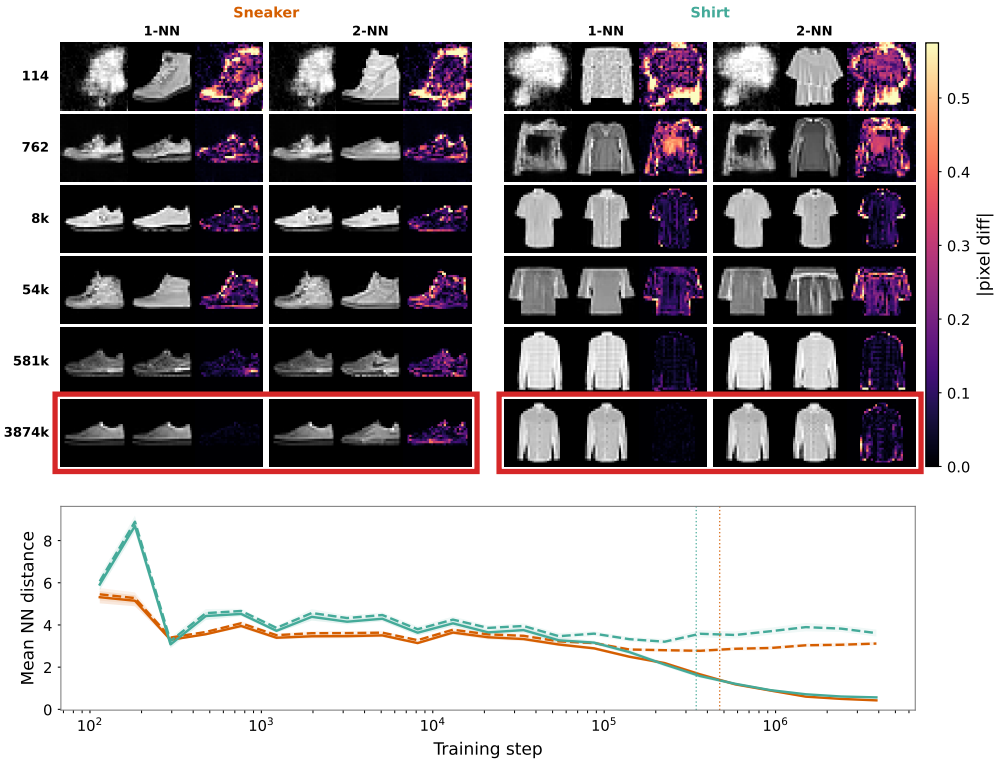
e Coat

Figure 14: Overview examples for Fashion MNIST classes (Part 5).



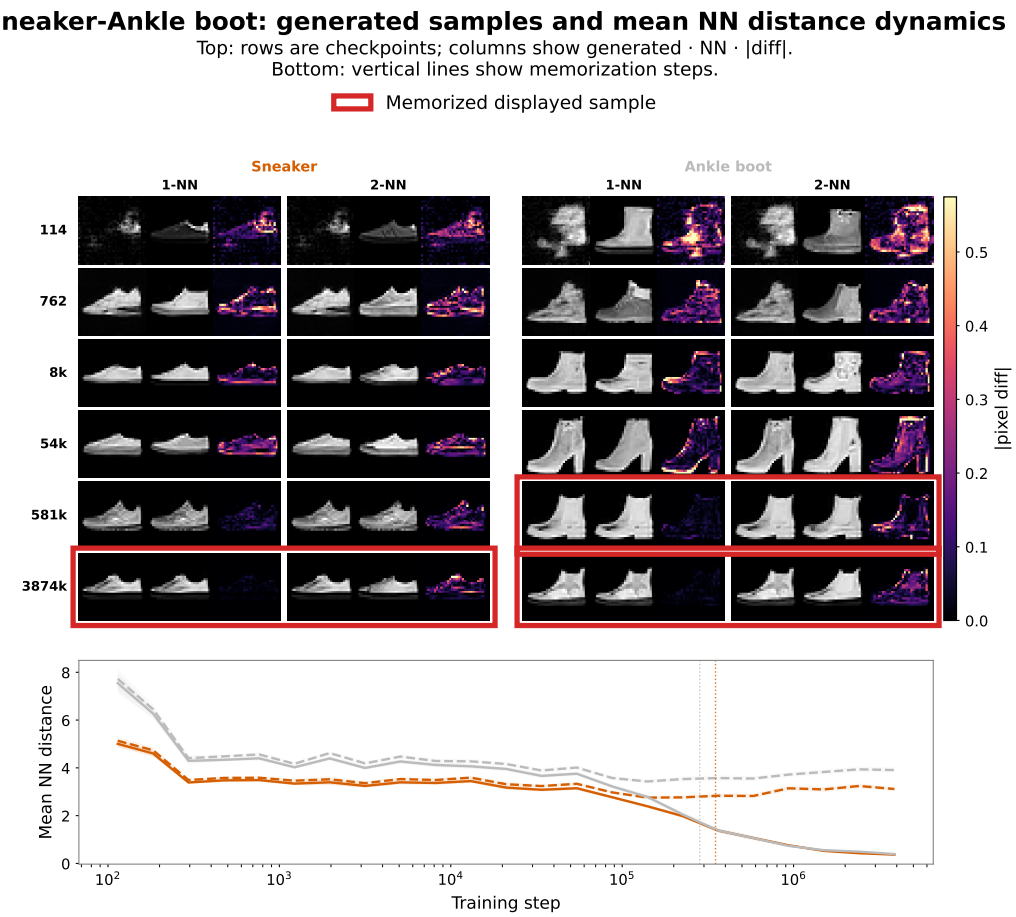
f Sandal

Figure 14: Overview examples for Fashion MNIST classes (Part 6).



g Shirt

Figure 14: Overview examples for Fashion MNIST classes (Part 7).



h Ankle boot

Figure 14: Overview examples for Fashion MNIST classes (Part 8).

Memorization fraction curves. We next extend the illustration in Fig. 5(a) by showing the memorization fraction curves for all Sneaker-paired Fashion MNIST classes in Fig. 14. Including early checkpoints shows that the memorization fraction remains zero for all classes before the onset of memorization. Afterward, all paired classes except Trouser reach the memorization threshold before Sneaker.

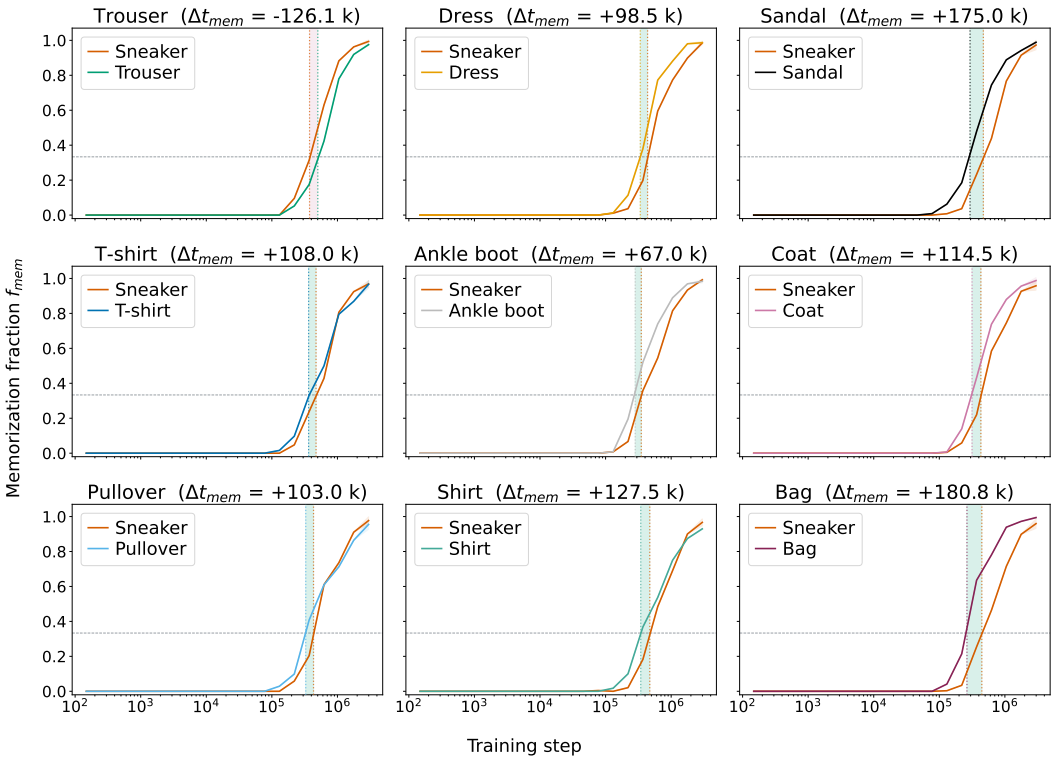


Figure 15: Memorization fraction curves for all Sneaker-paired Fashion MNIST classes.

Further validation on CIFAR. We further present the same phenomenon on CIFAR [29] using the same DDPM architecture and training settings, with 3000 training samples. The left panel of Fig. 16 shows that CIFAR classes have smaller normalized in-class variances than Fashion MNIST classes. Nevertheless, when zooming in on the lowest- and highest-variance CIFAR classes, Automobile is memorized before Deer, matching the variance-based prediction. Absolute memorization times differ from those on Fashion MNIST, which is expected because the CIFAR experiment uses a different total sample size; this is consistent with the theoretical prediction that sample number also affects memorization time [8, 27]. The corresponding generated samples and the dynamics of the nearest-neighbor distance are shown in Fig. 17.

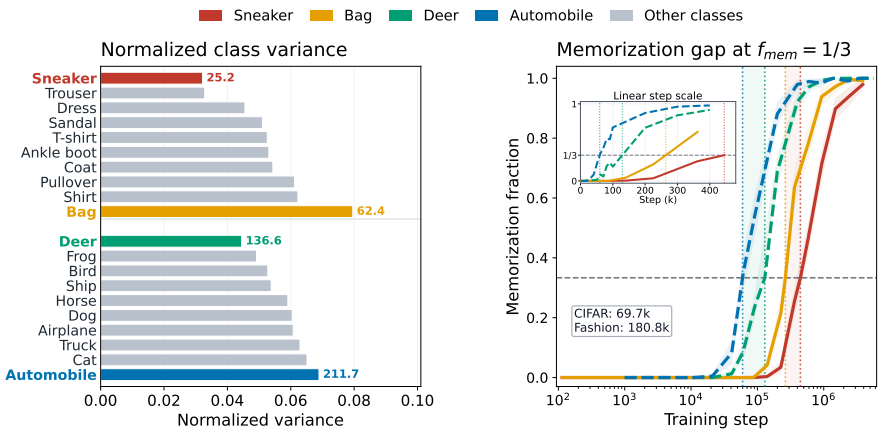


Figure 16: Left: normalized variance for each class on two datasets; the annotated numbers report the total class variance. Right: memorization fraction curves for the class pair with the largest variance gap in each dataset. The inset shows the same curves on a linear training-step scale, and the legend reports the corresponding memorization-time gaps.

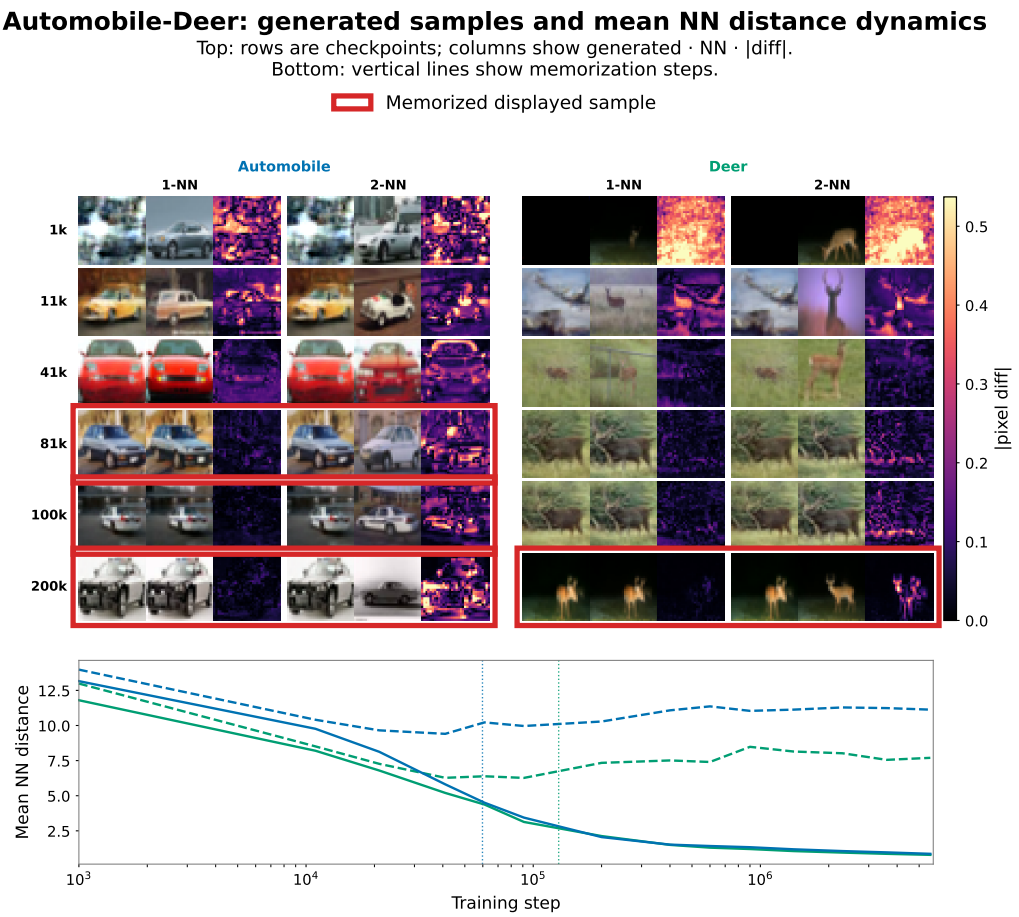


Figure 17: CIFAR generated samples and nearest-neighbor dynamics. For CIFAR, we restrict the second nearest neighbor to the same class. This avoids fluctuations in the average 2-NN distance caused by Automobile samples whose unconstrained second nearest neighbor may lie in Deer class, while preventing such fluctuations from shifting the Automobile memorization time earlier.

Robustness to memorization thresholds. Because memorization has no unique operational definition, choosing comparable thresholds across models is inherently difficult. We use $f_{\text{mem}} = 1/3$ for DDPM in the main text to capture a realistic regime where a substantial, but not exhaustive, fraction of samples is memorized. The same trends hold under stricter choices $f_{\text{mem}} = 1/6$ and $1/12$. For RF, whose criterion is tied to the initial loss, tightening the threshold from 20% worse than the initial loss to 10% worse, or equal to the initial loss, also preserves the qualitative picture (see Fig. 18 for DDPM and Fig. 19 for RF, respectively). Moreover, using RF threshold 1.3 gives a correlation with the DDPM result comparable to that obtained with the main threshold 1.2.

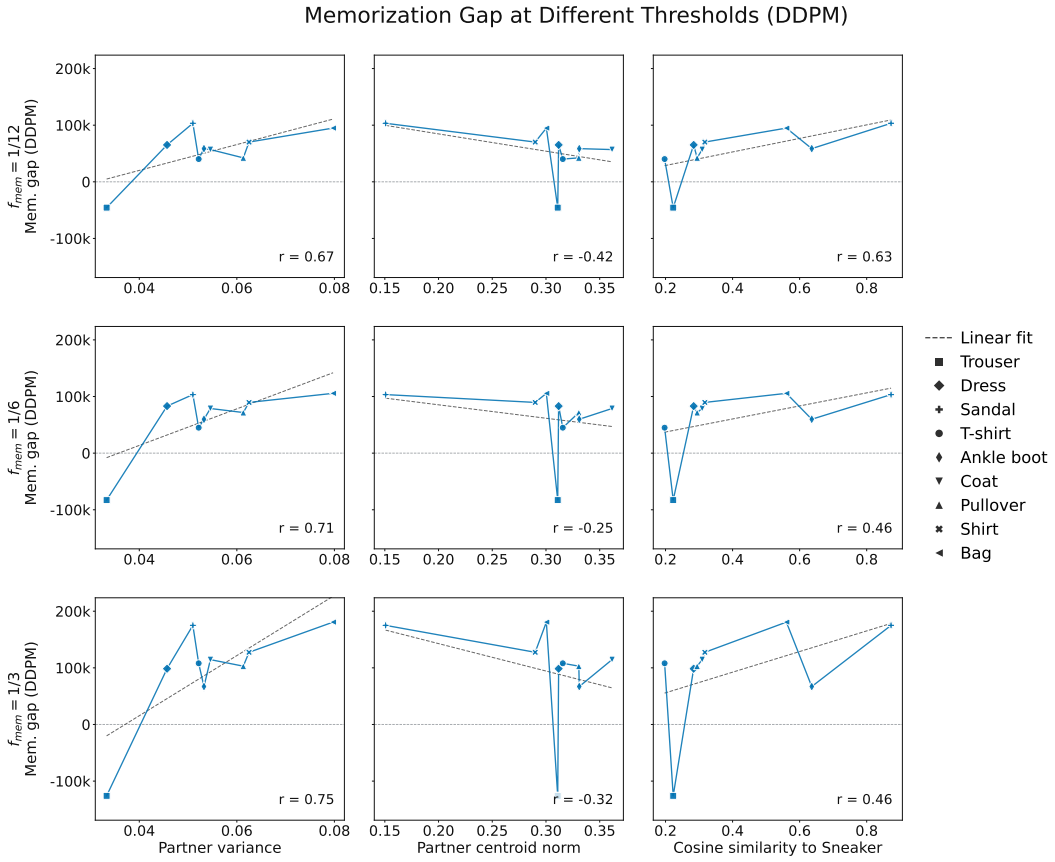


Figure 18: Same plot as Fig. 6, but using alternative thresholds to compute the DDPM memorization gap Δt_{mem}. The three rows correspond to $f_{\text{mem}} = 1/12$, $1/6$, and $1/3$, with the definition of Δt_{mem} given in Section 4. The overall patterns are similar across these threshold choices.

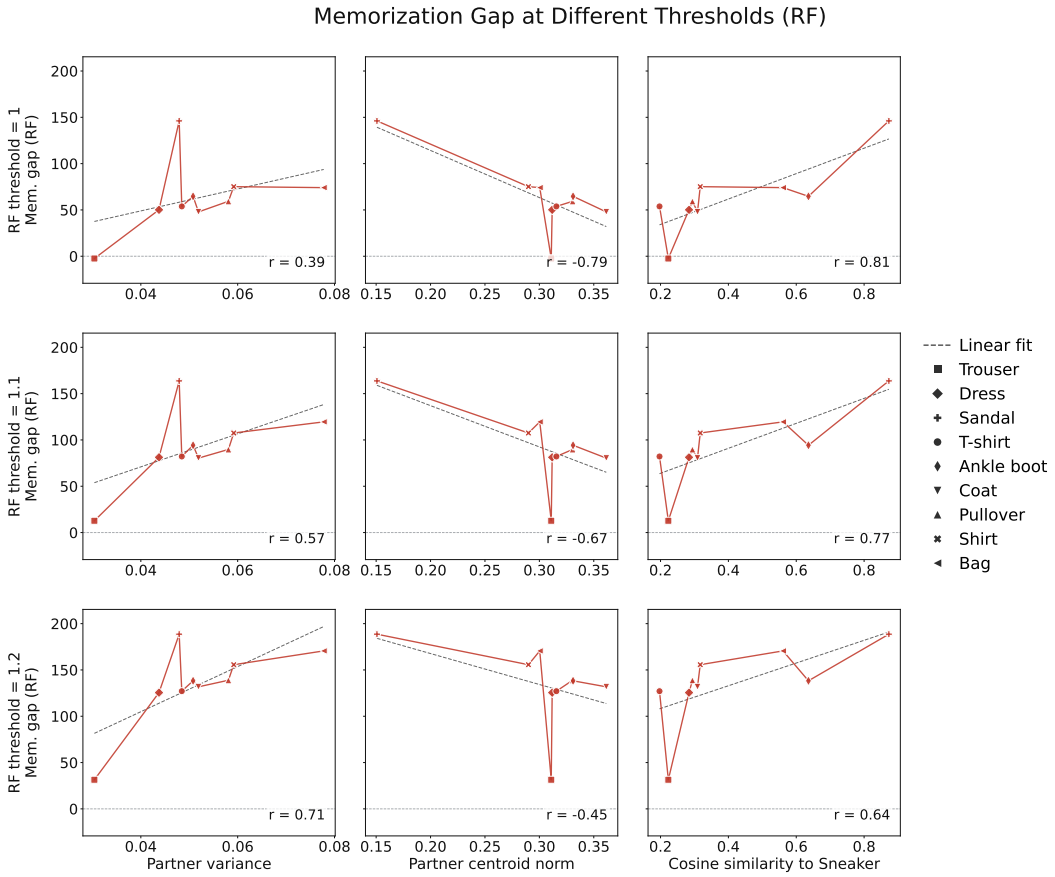


Figure 19: Same plot as Fig. 6, but using alternative RF memorization thresholds. The RF threshold is defined by how much the test loss is allowed to degrade relative to the initial loss, as described in Section 3.2.1; for example, a threshold of 1.1 means 10% worse than the initial loss. Similar patterns persist across thresholds.

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 reported in the abstract are based on the results outlined in Section 3 and Section 4. As delineated in the Section 1, the scope of this work is contextualized with pertinent references in the field. Readers interested in a more detailed treatment of this topic are referred to the Appendix B.

   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: The paper discusses limitations in the conclusion, including the scope of the theoretical assumptions. These limitations also point to meaningful directions for future research.

   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: The assumptions behind theoretical results are outlined in the 'Problem Formulation' Section 2. Proofs of the theoretical results are outlined in Appendices, in the theory section.

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: The paper discloses the information needed to reproduce the main experimental results. Readers can find the relevant details in the appendix, including the datasets, RF model construction, diffusion-model architecture and training setup, classifier-based class assignment, algorithms, preprocessing, hyperparameters, and evaluation procedures. The experiments follow standard practice and use standard public datasets and model classes, providing a clear path for reproducing the results.

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: [No]

   Justification: Presently, the codes are unavailable. However, the main text and the supplemental material provide the necessary methodological and implementation details to faithfully reproduce the main experimental results. We are committed to releasing the data and code publicly upon acceptance.

   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: Readers can find the necessary details for understanding and reproducing the numerical experiments in the appendix. This includes the RF model experiments, diffusion model (DDPM) experiments, classifier-based class assignment, and related algorithms, together with data splits, model architectures, optimizers, hyperparameters, preprocessing, and implementation choices.

   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: The paper reports error bars or confidence intervals for experiments involving averages or repeated measurements. For example, we performed bootstrapping, and the figure captions explain how the uncertainty estimates are computed.

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: The paper provides the details needed to reproduce the experiments, with additional information reported in the experimental appendix. This includes the compute setup, execution details, model architectures, training hyperparameters, and other experimental parameters for the reported runs.

   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: The research conforms to the NeurIPS Code of Ethics. The paper is theoretical and uses standard benchmark datasets and synthetically generated data, without crowdsourcing, human-subject experiments, scraped datasets, or high-risk model releases.

   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: [N/A]

    Justification: The paper is primarily theoretical and is not focused on a specific deployment or application domain. While the conclusion discusses practical implications of higher-variance, small-centroid, and highly imbalanced data—including how such heterogeneity may lead to unreliable or unfair behavior across groups or contexts—a rigorous analysis of mitigation strategies is beyond the scope of this work. We view such analysis as an important and interesting direction for future work, rather than a main theme of the paper.

    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 does not release high-risk models or datasets, such as pre-trained language models, large-scale image generators, or scraped datasets. The experiments use standard benchmark datasets and only generate samples from Fashion-MNIST and CIFAR, which do not pose the misuse risks described in this question.

    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: The paper uses only well-known public benchmark datasets and standard model classes. We cite the original papers for all datasets used, including Fashion-MNIST and CIFAR, and cite relevant prior work when our implementation is inspired by existing code or model designs with all important experimental parameters are reported in the appendix. We respect the corresponding licenses and terms of use for these existing assets.

    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: [N/A]

    Justification: The paper does not introduce new datasets, pretrained models, or benchmark assets. The experiments use standard benchmark datasets, including Fashion-MNIST and CIFAR, and standard model classes such as DDPMs. The implementation code will be released upon acceptance and documented separately.

    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, participant studies, or any interaction with human subjects. All experiments are conducted on standard benchmark datasets or synthetically generated data, so participant instructions, screenshots, and compensation details are not applicable.

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 is theoretical and includes numerical experiments only on standard benchmark datasets and synthetically generated data. It does not involve crowdsourcing, interaction with human subjects, or collection of new human-subject data; therefore, IRB approval is not applicable.

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: LLMs were not used as an important, original, or non-standard component of the core methodology. They were used only for writing and editing assistance, such as grammar, wording, formatting, and code-editing suggestions, including requests to the Google Paper Assistant Tool (PAT). The authors are responsible for the correctness, originality, and scientific content of the paper, including all text, figures, references, and code.

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.