# Generalization Dynamics of Linear Diffusion Models

**Claudia Merger** **Sebastian Goldt**
cmerger@sissa.it sgoldt@sissa.it
Theoretical and Scientific Data Science Group
SISSA
Trieste, Italy

## Abstract

Diffusion models are powerful generative models that produce high-quality samples from complex data. While their infinite-data behavior is well understood, their generalization with finite data remains less clear. Classical learning theory predicts that generalization occurs at a sample complexity that is exponential in the dimension, far exceeding practical needs. We address this gap by analyzing diffusion models through the lens of data covariance spectra, which often follow power-law decays, reflecting the structure of real data. To understand whether such a power-law structure can benefit learning in diffusion models, we develop a theoretical framework based on linear neural networks, congruent with a Gaussian hypothesis on the data. We quantify how the covariance spectra of data and regularization impact generalization. We find two regimes: When $N < d$, not all directions of variation are present in the training data, which results in a large gap between training and test loss. In this regime, we demonstrate how a power-law data structure, as well as regularization and early stopping help to prevent overfitting. For $N \gg d$, we find that the sampling distributions of linear diffusion models approach their optimum (measured by the Kullback-Leibler divergence) linearly with $d/N$, independent of the specifics of the data distribution. Our work clarifies how sample complexity governs generalization in a simple model of diffusion-based generative models.

Diffusion models Sohl-Dickstein et al., Song and Ermon [2019], Ho et al. [2020] are trained to sample from an unknown distribution $\rho$ using only a finite dataset. Their success suggests that, in practice, the relevant structure of $\rho$ can be learned from far fewer samples than would be required to approximate an arbitrary function in $d$ dimensions. Understanding which structure enables this finite-sample generalization remains an open problem.

Existing theory has largely focused either on diffusion models that have already learned $\rho$ accurately De Bortoli et al. [2021], Block et al. [2022], Bortoli [2023], Liu et al. [2022], Lee et al., Pidstrigach [2022] or on learning dynamics for unstructured Gaussian data George et al. [2025b,a], Bonnaire et al. [2025]. Much less is known about diffusion models trained on finite, structured datasets.

We focus on a simple, directly measurable form of structure: the covariance spectrum of the data. Covariance spectra, i.e. the distribution of the eigenvalues $\lambda_\nu$ of the covariance matrix $\Sigma$ of the data distribution, inform us about the relative spread of the data in different directions. For example, the covariance spectra of image data typically exhibit a power-law behavior, see fig. 1 a), meaning that the eigenvalues of $\Sigma$ behave as

$$\lambda_\nu \sim \nu^{-k}, \tag{1}$$

where $\nu$ is the eigenvalue rank and $k \geq 0$ controls the rate of spectral decay. This decay in the eigenvalues implies that the few leading eigen-directions of the covariance matrix account for the bulk of the variability in the data. The corresponding eigenvectors are often long-range features of the data, for example controlling background color or the placement of shadows in an image, see fig. 1 e)-g), while sub-leading eigenvectors correspond to more detailed features. In the case when $\Sigma$ is exactly known ($N \to \infty$), covariance spectra are known to affect learning dynamics in diffusion

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

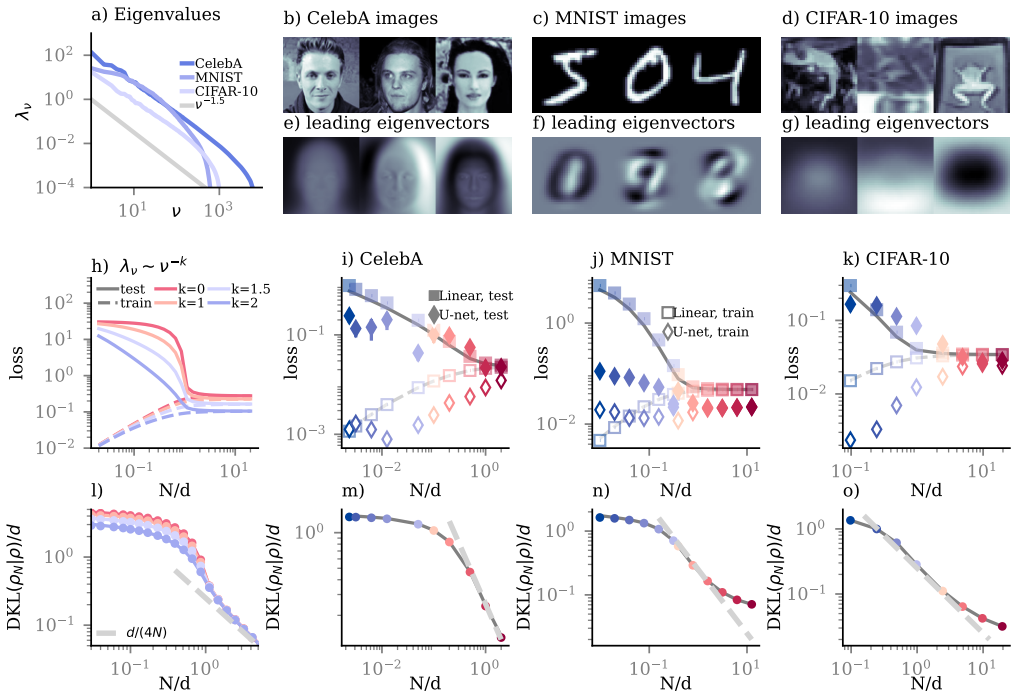



Figure 1: a) Eigenvalues of covariances obtained from image data sets, sorted by rank. b) - d) Example images from image datasets. e) - g) Top three leading eigenvectors of covariance matrices obtained from the full image dataset. h) Prediction for test and training loss of linear diffusion models trained on $N$ samples $d = 100$-dimensional samples from $\mathcal{N}(0, \Sigma)$, where the eigenvalues of $\Sigma$ follow a powerlaw with exponent $k$ normed such that $\mathrm{Tr}\Sigma/d = 1$. i) - k) Test and train loss of trained diffusion models with linear and U-net architecture trained on $N$ training data. Test losses are averaged over $10^4$ samples from the test set. Training losses are computed using at $\max(N, 10^4)$ training data. Grey lines show prediction from replica theory. l) Kullback-Leibler divergence between sample distribution of linear diffusion models with regularization $c = 10^{-4}$ and $\mathcal{N}(0, \Sigma)$, where the eigenvalues of $\Sigma$ follow the same powerlaw as in h). Symbols are averages over 10 random draws of the training sets, error bars report one standard deviation, but are typically smaller than the symbol size. m) - o) are equivalent to l), but for $\Sigma$ originating from the CelebA, MNIST, and CIFAR-10 datasets, respectively. Grey lines show prediction eq. (7) from replica theory.

models: directions with larger eigenvalues are typically learned faster than sub-leading ones Catania et al. [2025], Wang and Pehlevan.

In this work, we investigate how power-law covariance spectra affect learning in diffusion models when the number of data points $N$ is finite. In this case, we only have access to $\Sigma$ through a noisy estimate $\Sigma_0$ determined from the data, which grows more accurate with $N$. Moreover, leading eigendirections of $\Sigma$ can typically be recovered at smaller sample complexity, meaning that a strong decay in the spectrum of $\Sigma$ facilitates estimating leading eigendirections from $\Sigma_0$. Here, we investigate how the interplay of the sample complexity $N$, the decay $k$ and the training dynamics affects generalization dynamics in diffusion models. To this end, we will consider a linear neural network that is able to produce samples with arbitrary covariance. Our main contributions are

- For $N < d$, the empirical covariance $\Sigma_0$ has a nontrivial nullspace, which drives overfitting; stronger spectral decay in $\Sigma$ mitigates this effect.
- Regularization acts as an effective covariance cutoff and mitigates overfitting; the optimal regularization strength decreases with both $N$ and $k$.
- Larger spectral decay slows learning and delays overfitting, creating a broader window in which early stopping improves generalization.

- Changing the standard denoising diffusion objective from noise prediction to data prediction *reweights* the test loss toward the leading eigenmodes, especially as $k$ increases.

Our theoretical results are complemented by several numerical experiments comparing the behavior of linear models to their nonlinear counterparts.

**Related Work.** Several theoretical approaches explain diffusion-model generalization by imposing structure on the data distribution. One line studies compositional structure, such as the random hierarchy model, where generalization can occur when $N$ scales polynomially with $d$ Cagnetta et al. [2024], Favero et al. [2025b,a]. Another assumes that data lie on a low-dimensional manifold, so that sample complexity depends on intrinsic rather than ambient dimension Chen et al. [2023], Achilli et al. [2025], Wang et al. [2024], Goldt et al. [2020]. These approaches identify mechanisms by which high-dimensional distributions can be learnable, but they do not directly yield quantitative predictions from the measured spectrum of a given dataset; moreover, recent work suggests a more refined notion of image dimensionality than a homogeneous low-dimensional manifold Guth et al. [2025]. When the population covariance $\Sigma$ is known, covariance spectra are known to induce spectral bias in diffusion-model learning dynamics, with high-variance directions learned earlier Catania et al. [2025], Wang and Pehlevan. We study the finite-sample setting, where $\Sigma$ is observed only through $\Sigma_0$, using a linear diffusion model in the spirit of Biroli and Mézard [2023], as the simplest model capable of learning such structure. Linear neural networks have helped elucidate overfitting in supervised learning in the past Advani et al. [2020], Fischer et al. [2022] by providing a fully tractable case in which key mechanisms can be understood.

## 1 Linear Diffusion Models

We now give a brief introduction to diffusion models before explaining our approach. Diffusion models consist of an iterative noising and denoising process. The noising process simplifies the distribution of a sample $x_0$ from $\rho$ through the addition of noise

$$x_t\left(\epsilon_t\right) = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon_t, \qquad \epsilon_t \sim \mathcal{N}\left(0, \mathrm{Id}\right) , \tag{2}$$

for a number of noising steps $t \in \{1, \dots, T\}$ and $\bar{\alpha}_t \in (0,1)$ is a decreasing function in $t$. As $t$ increases, the original signal $x_0$ is gradually suppressed compared to the isotropic Gaussian noise, until one obtains $x_T$ whose distribution is close to $\mathcal{N}\left(0, \mathrm{Id}\right)$. Diffusion models are neural networks $\epsilon_\theta$ whose parameters $\theta$ are then optimized to approximate the score $\epsilon_\theta(x_t, t) \propto \nabla \ln \rho_t(x_t)$ , where $\rho_t$ is the density of the noised variables $x_t$. This is achieved via the denoising score matching objective

$$L = \frac{1}{dTN} \sum_t \sum_{x_0 \in \mathcal{D}} \mathbb{E}_{\epsilon_t} \left\| \epsilon_t - \epsilon_\theta\left(x_t\left(\epsilon_t\right), t\right) \right\|^2 , \tag{3}$$

where $\mathcal{D}$ is a training set of size $N$ drawn i.i.d. from $\rho$.

Typical architectures for $\epsilon_\theta$ are very complex, including U-nets Ronneberger et al. [2015] and transformers. Here, we consider the case where for each $t$, the denoiser $\epsilon_\theta(\cdot, t)$ is an affine linear mapping, whose weights we additionally regularize using a standard $L^2$ penalty whose strength $\gamma_t$ depends on $t$. Once trained, the denoiser is then used to iteratively generate new samples. We provide details on the training and generation process in appendix A. The learning outcomes of the linear model can be expressed using only the empirical mean $\mu_0$ and covariance $\Sigma_0$ of the training set $\mathcal{D}$ and the population mean $\mu$ and covariance $\Sigma$ of $\rho$, see appendix A. When the number of training samples, $N$, is finite, the empirical mean and covariance will deviate from their population averages $\mu_0 \neq \mu$, $\Sigma_0 \neq \Sigma$. The central aim of our work is to predict how this mismatch affects the diffusion model's ability to generalize to new data depending on $k$.

## 2 Power-law spectra and overfitting

**The nullspace of $\Sigma_0$ drives overfitting.** As a first measure for generalization, we now compare the minimal training loss $R$ to the test loss $L_{\text{test}}$, which quantifies the ability of the denoiser to denoise an unseen test example from $\rho$. We express this using the eigendecomposition of the empirical covariance matrix $\Sigma_0$, with eigenvalues $\lambda_\nu^0$, and normalized eigenvectors $\{e_\nu^0\}_\nu$. The gap between the $R$ and $L_{\text{test}}$ is

$$L_{\text{test}} - R = \sum_t \left(\bar{\alpha}_t - \bar{\alpha}_t^2\right) \sum_\nu \frac{\left(e_\nu^0\right)^{\mathrm{T}} \Sigma\, e_\nu^0 - \lambda_\nu^0 + (\mu - \mu_0)_\nu^2}{\left(\bar{\alpha}_t \lambda_\nu^0 + (1 - \bar{\alpha}_t + \gamma_t)\right)^2} . \tag{4}$$

This shows explicitly that the gap between $R$ and $L_{\text{test}}$ arises whenever there is a mismatch between $\mu_0$, $\Sigma_0$ and $\mu$, $\Sigma$. The terms that contribute the most strongly to $L_{\text{test}} - R$ are those for which the denominator under the sum is minimized. This occurs for the smallest values of $\lambda_\nu^0$ and the smallest values of $t$, as there $1 - \bar{\alpha}_t$ is minimized. When the number of data in the training set, $N$, is smaller than the dimension $d$, at least $d - N$ eigenvalues of $\Sigma_0$ are exactly zero, giving rise to large contributions in eq. (4). This is reflected in a very large gap between training and test loss in fig. 1 h-k) when $N < d$. However, this gap can effectively be reduced by regularization, through the presence of $\gamma_t$ in the denominator of eq. (4). We compare our results using linear models to U-nets trained on image data in fig. 1 h) - j). Remarkably, these highly non-linear models both reproduce the large gap at $N < d$ and a similar saturation at $N \approx d$. For large $N >> d$, U-nets naturally outperform their linear counterparts.

We compute Kullback-Leibler divergence (DKL) comparing the distributions of the generated samples $\rho_N$, and $\rho$. For linear denoisers, $\rho_N = \mathcal{N}(\mu_0, \Sigma_0 + c\text{Id})$, where $c$ is a small parameter. The presence of $c$ can be interpreted as originating either from the corrections due to the finite number of sampling steps , or from the regularizing with a particular choice $\gamma_t = \bar{\alpha}_t c$ for the regularization strength (see appendix B for detailed derivation).

We now impose a Gaussian hypothesis on the data $\rho = \mathcal{N}(\mu, \Sigma)$ to study how the spectral decay in $\Sigma$ controls the deviations between $\rho, \rho_N$ which arise due to finite $N$. The DKL is then

$$\text{DKL}(\rho_N|\rho) = \frac{1}{2}\left[\ln\frac{|\Sigma|}{|\Sigma_0 + c\text{Id}|} + (\mu - \mu_0)^{\text{T}}\Sigma^{-1}(\mu - \mu_0) + \text{Tr}\Sigma^{-1}(\Sigma_0 + c\text{Id}) - d\right], \tag{5}$$

which is a measure of distance between distributions. The most dominant term at small $N$ in $\text{DKL}(\rho_N|\rho)$ is $\ln\frac{|\Sigma|}{|\Sigma_0^{\text{eff.}}|} = \sum_\nu \ln\frac{\lambda_\nu}{\lambda_\nu^0 + c}$, where $\lambda_\nu$ are the eigenvalues of $\Sigma$. This term in the DKL heavily penalizes the presence of a nullspace in $\Sigma_0$ for small $c$, analogous to the test loss, but also indicates that the penalty for a nullspace of $\Sigma_0$ is less severe when $\Sigma$ has small eigenvalues, hence when its spectrum decays faster.

**Powerlaw spectra mitigate overfitting.** So far, our measures of generalization depend on the specific realization of $\Sigma_0$, computed from one draw of the training set.

We use the replica trick to evaluate the necessary averages; the full calculation is found in appendix E. To capture the effect of a decay in the spectrum of $\Sigma$ explicitly, we now average over draws of the training set from which $\Sigma_0$ is computed. For this average, we assume that the data are drawn from a centered Gaussian and replace $\Sigma_0$ by the empirical second moment. The latter is a Wishart matrix with $N$ degrees of freedom, whereas $\Sigma_0$ covariance has $N - 1$ degrees of freedom. We neglect this finite-$N$ distinction here, since we work in the proportional limit $N, d \to \infty$ at fixed $N/d$.

For brevity, we will focus on the typical case analysis of the DKL here. Our results depend on a quantity $q$, defined by

$$q = \frac{c}{d}\text{Tr}\left\langle \Sigma \frac{1}{\text{Id}c + \Sigma_0}\right\rangle \tag{6}$$

where the expectation is over realizations of $\Sigma_0$. It can therefore be interpreted as the amount of population variance that remains unresolved by the empirical covariance at the relevant regularization scale. In the average over draws of the training set, the DKL is given by

$$\begin{aligned}\frac{1}{d}\text{DKL}(\rho_N|\rho) =& \frac{1}{2}\frac{q}{\frac{d}{N}q + c} - \frac{1}{2d}\sum_i \ln\left|\frac{c}{\lambda_i} + \frac{1}{\frac{d}{Nc}q + 1}\right| - \frac{N}{2d}\ln\left(\frac{d}{Nc}q + 1\right) \\ &+ \frac{d + 2\sqrt{c}\text{Tr}\Sigma^{-\frac{1}{2}} + c(N+1)\text{Tr}\Sigma^{-1}}{2Nd}\end{aligned} \tag{7}$$

In fig. 1 l)-o) we compare the prediction obtained from eq. (7) to numerical simulations, showing excellent agreement. The first line in eq. (7) originate from the term $\ln|\Sigma|/|\Sigma_0 + c\text{Id}|$ and dominate the expression when $c$ is very small. These terms diminish with $q$, hence for smaller $q$, we find that overfitting is less severe at fixed $N/d$, therefore $q$ here governs the overfitting behavior.

In appendix E, we show that $q$ can be determined self-consistently from the eigenvalues $\{\lambda_\nu\}_{\nu=1}^d$ of $\Sigma$ and $c$ using

$$q = \frac{1}{d}\sum_\nu \frac{\lambda_\nu}{1 + \lambda_\nu \frac{N}{dq + Nc}} \tag{8}$$

Importantly, this implies that (see appendix E.8 for a derivation)

$$q \leq \left(\frac{d}{N}\bar{\lambda} + c\right) \; \frac{1}{d}\mathrm{Tr}\frac{\Sigma}{\Sigma + \mathrm{Id}\frac{d}{N}\left(\bar{\lambda}+c\right)} \tag{9}$$

where $\bar{\lambda} = \frac{1}{d}\mathrm{Tr}\Sigma$ is the average over the eigenvalues of $\Sigma$, and we use $\frac{1}{A}$ to denote the matrix inverse of $A$. At fixed $d/N$, the faster the spectrum decays, i.e. the more eigenvalues of $\Sigma$ are significantly smaller than average, the smaller $q$ will be, and therefore, the smaller the DKL. This is evidently true for the upper bound on $q$, for the full self-consistency equation we prove it in appendix E.8.2. We also show examples of this both for spectra which are explicitly powerlaw, $\lambda_\nu \sim \nu^{-k}$ and $\Sigma$ determined from image data in fig. 1 l)- o). Similar results hold for the gap between test and training loss, shown in the same setting in 1 h)-k). This demonstrates how a faster decaying spectrum can lead to a better fit. Intuitively, this effect of the spectral decay in the $N < d$ regime can be explained as follows: since the absence of variation in $\Sigma_0$ is not as significant when the corresponding variation in $\Sigma$ is also small. The theoretical calculations performed here turn this qualitative picture quantitative, as we can use $q$ to measure how much the nullspace of $\Sigma_0$ aligns with the low-variance eigenspace of $\Sigma$.

On the other hand, for $N > d$, the DKL collapses on to the same line independently of the specifics of $\Sigma$. The independence of the DKL on $\Sigma$ has been noted in Tumminello et al. [2007], who argued that this makes the DKL a good measure for the similarity of $\Sigma_0$ to $\Sigma$. In appendix E.8.3, we show that when $c$ is much smaller than the smallest eigenvalue of $\Sigma$ and $N > d$ the DKL is approximately given by $d/(4N)$, where we have neglected terms of order $(d/N)^2$ and $\sqrt{c}$. In the $N > d$ regime, we find this scaling of the DKL across realizations of $\Sigma$ (see fig. 1 l) - o)), up to deviations which originate from $c > 0$, which causes a saturation of the DKL above zero. Therefore, the powerlaw-decay affects the DKL as a performance measured predominantly in the $N < d$ regime, and not vice versa.

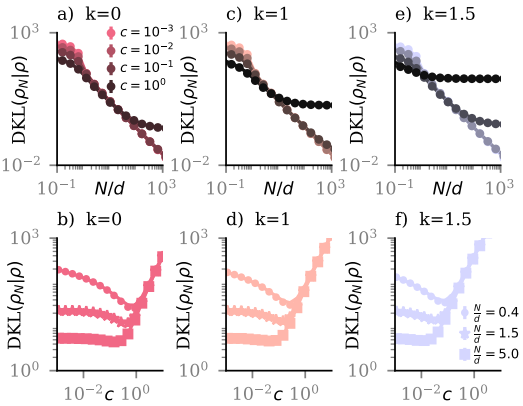



Figure 2: a)-b) Kullback-Leibler divergence between sample distribution of linear diffusion models from Gaussian ground truth for varying levels of regularization number of data drawn from $\mathcal{N}(0, \mathrm{Id})$. c)-f) are equivalent to a), b) but for data drawn from and $\mathcal{N}(0, \Sigma)$, where the eigenvalues of $\Sigma$ follow a powerlaw with $\lambda_\nu \sim \nu^{-k}$. Symbols are averages over 10 random draws of the training sets, error bars report one standard deviation, but are typically smaller than the symbol size. Lines show prediction eq. (7) from replica theory.

**Regularizing places a spherical bias on the covariance.** For each timestep $t$, regularizing the linear model with strength $\gamma_t$ effectively replaces $\bar{\alpha}\Sigma_0$ with an effective covariance matrix $\bar{\alpha}_t\Sigma_0 + \gamma_t\mathrm{Id}$. Here, we have chosen to regularize earlier timesteps (which overfit more strongly) more heavily than later timesteps by setting $\gamma_t = \bar{\alpha}_t c$. This results in an effective data covariance matrix $\Sigma_0 \to \Sigma_0 + c\mathrm{Id}$. $c$ therefore represents an effective cutoff on the eigenvalues of $\Sigma_0$, because those directions where $\lambda_\nu^0 \ll c$ will no longer be distinguishable from each other. Hence in the undersampled regime, one can find an optimal $c > 0$ as a function of the spectral decay exponent $k$ and $d/N$.

To illustrate this point, we show the DKL of linear diffusion models trained with different regularization strengths on increasing number of data in fig. 2 a)-c). For small $N$, regularizing has a beneficial effect. At larger $N$, too much regularization eventually obstructs the structure in $\Sigma$ that could otherwise be resolved via $\Sigma_0$, leading to an increase in the DKL. When the spectrum decays faster, a larger number of eigenvalues is masked by the same level of $c$. This leads to an increase in the DKL between the regularized linear model and the optimal linear model. At fixed, finite number of data, increasing the level of regularization lowers the DKL (improves generalization) up to a certain point, beyond which it is detrimental (see lower row fig. 2). The optimal level of regularization decreases with the spectral decay $k$, concurrent with our intuition that $c$ masks the subspace $\Sigma_0$ with smaller eigenvalues, which is larger for power-law data.

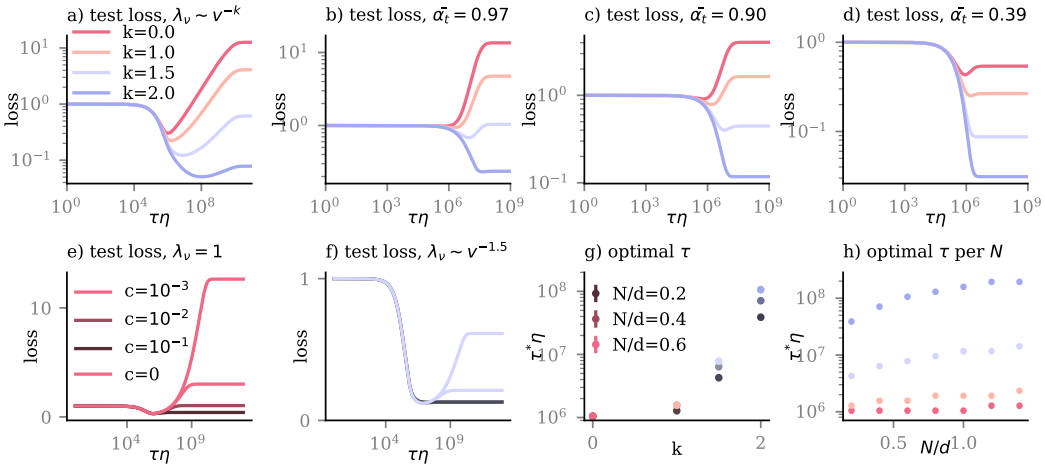



Figure 3: a) Test loss of linear diffusion models trained on $N = 0.6d$ samples from a centered Gaussian with powerlaw eigenvalues, $\lambda_\nu \sim \nu^{-k}$. b-d) are equivalent to a), but for fixed level of noise, $t$. e)-f) are equivalent to a), but for linear denoisers trained with increasing regularization strength $c$. g) Training time $\tau$ with optimal test loss as a function of $k$ for different fractions $N/d$. h) same as g), but for different fractions $N/d$. All curves are averages over 5 draws of the data and $d = 10^3$.

**Regularization and early stopping prevent overfitting.** We now turn to an analysis of the dynamics of training diffusion models. To simplify the analysis, we assume that the data are centered, $\mu_0 = 0$ and we start from zero initial weights. Using full batch gradient descent with learning rate $\eta$, one finds that weights of the denoiser exponentially relax to their optima with a direction-and noise dependent rate

$$\eta_{\mu,t} = 2\frac{\bar{\alpha}_t \lambda_\mu^0 + (1 - \bar{\alpha}_t + \gamma_t)}{dT}, \tag{10}$$

see appendix C for a derivation. This means that leading eigen-directions of $\Sigma_0$ are learned faster than sub-leading ones at fixed $t$, hence the spectral bias observed in diffusion models trained on the population loss Wang and Pehlevan translates to diffusion models trained on finite data replacing $\Sigma \to \Sigma_0$. However, since we know which directions of $\Sigma_0$ are responsible for overfitting, we can now isolate their trajectories to understand how overfitting emerges dynamically during training. In fact, the gap between training and test loss evolves with the training time for $\tau$ as

$$\frac{d\left(L(\tau) - R(\tau)\right)}{d\tau} \propto \sum_{t,\mu} (1 - \bar{\alpha}_t)\left(1 - e^{-\eta_{\mu,t}\tau}\right)\left(\frac{\bar{\alpha}_t\left(1 - \left(e_\mu^0\right)^{\mathrm{T}} \Sigma\, e_\mu^0\right) + 1}{\bar{\alpha}_t \lambda_\mu^0 + (1 - \bar{\alpha}_t + \gamma_t)} - 1\right).$$

Starting from initially zero difference between these quantities, modes where $\left(e_\mu^0\right)^{\mathrm{T}} \Sigma\, e_\mu^0 > \lambda_\mu^0 + \gamma_t/\bar{\alpha}_t$ will make the gap between training and test loss widen over time. Regularization ($\gamma_t > 0$ ) can decrease the number of such modes and also shrink the gap. Since $\eta_{\mu,t} \sim \bar{\alpha}_t \lambda_\mu^0 + (1 - \bar{\alpha}_t + \gamma_t)$, the most precarious directions in the sense of overfitting are also the ones which are learned the slowest. This makes both early stopping and regularization effective strategies to prevent overfitting. We show the evolution of the test loss with training time $\tau$ in fig. 3a)-d), for specific levels of noise and in the average over all $t$. We find that the optimal $\tau^*$ where the test loss is minimized increases as the spectrum decays (see fig. 3g)) and as we increase $N$ (see fig. 3h)). This means both that more faster-decaying spectra require more training, but also overfit later. Regularization prevents overfitting: In fig. 3 e), f) we apply regularization with strength $\gamma_t = c\bar{\alpha}_t$ to the weights of the model. We see that even moderate levels of regularization can decrease the test loss drastically.

**Predicting the data instead re-weights the objective.** We now investigate whether the learning problem changes if instead of predicting the noise, one aims to predict the original data. In principle, both objectives are equivalent, since, if one has a model $\epsilon_\theta(x_t, t)$ that predicts the noise, one can use it to define a function $f_\theta(x_t, t)$ to predict the data, using that eq. (2) implies a linear relation between

the two

$$f_\theta^*(x_t, t) = \frac{x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta^*(x_t, t)}{\sqrt{\bar{\alpha}_t}}\,. \tag{11}$$

However, we can also choose to directly learn $f_\theta$ with the corresponding objective

$$L_D = \frac{1}{dTN}\sum_t \sum_{x_0 \in \mathcal{D}} \mathbb{E}_{\epsilon_t} \left\| x_0 - f_\theta\left(x_t\left(\epsilon_t\right), t\right)\right\|^2, \tag{12}$$

where we use the subscript $D$ to signify we are predicting the data instead of the noise. It is straightforward to show that the optimal affine linear counterpart $f_\theta^*$ of $\epsilon_\theta^*$ fulfills eq. (11). Moreover, linear models trained to convergence with either objective have the same Kullback-Leibler divergence and learn modes with the same speed; as we show in appendix D. Consequently, using eq. (12) on data rather than eq. (3) represents a re-weighting of the terms in the loss. Indeed, given a matching initialization, we find that $f_\theta, \epsilon_\theta$ fulfill eq. (11) at any training stage $\tau$.

This re-weighting of the terms in the test loss is especially significant in the presence of zero modes in $\Sigma_0$. Indeed, we find that the gap between test and training loss for this objective is

$$\begin{aligned} L_{\text{test},D} - R_D =& \operatorname{Tr}\left(\Sigma - \Sigma_0\right) \\ &+ \frac{1}{dT}\sum_{t,\mu}\left[2\left(1-\bar{\alpha}_t\right) - \lambda_\mu^0 \bar{\alpha}_t\right] \\ &\cdot \frac{\bar{\alpha}_t \lambda_\mu^0 \left(\left(e_\mu^0\right)^{\mathrm{T}} \Sigma e_\mu^0 - \lambda_\mu^0\right)}{\left(\bar{\alpha}_t \lambda_\nu^0 + 1 - \bar{\alpha}_t\right)^2}\,. \end{aligned} \tag{13}$$

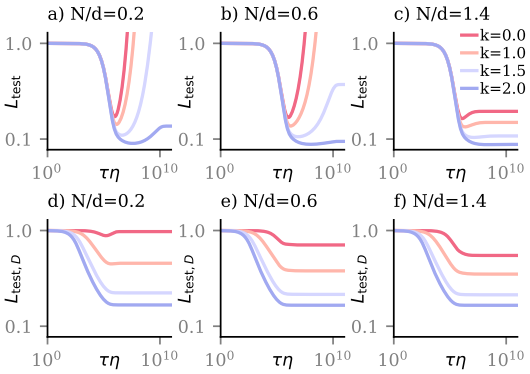



Figure 4: a)-c) Standard test loss of linear diffusion models eq. (3), trained on increasing fractions of data as a function of training steps $\tau$ times learning rate $\eta$. d)-f) Same as a)-c), but for test loss corresponding to predicting the data instead of the noise, eq. (12). Data are drawn from a centered Gaussian with powerlaw eigenvalues, $\lambda_\nu \sim \nu^{-k}$. All curves are averages over 5 draws of the data and $d = 10^3$.

where we have neglected terms proportional to the difference of the means $\mu - \mu_0$ for brevity. Here, we see that the terms corresponding to the nullspace of $\Sigma_0$ drop out in the second term, due to the factor $\lambda_\mu^0$ in the numerator. Rather, the nullspace of $\Sigma_0$ only contributes to gap between test and training loss via the first term $= \operatorname{Tr}\left(\Sigma - \Sigma_0\right)$. Consequently, the objective $L_D$ places comparatively lower emphasis on learning all directions equally well than the objective learning the noise, $L$ (eq. (3)). We show the difference between the two objectives in fig. 4 for three different dataset sizes. When $k$ is large, we find that $L_D$ hardly distinguishes between the three cases, whereas we can see a large difference in $L$ and the DKL (see fig. 1 l)). For $k = 0$, overfitting can be seen by a dynamical decrease and subsequent increase in both objectives when the number of data is small $N/d = 0.2$ but is far less pronounced in $L_D$ than in $L$. Thus we see that in the low data regime, $L$ is more representative of the DKL than $L_D$.

## 3 Comparison between linear and non-linear models

In the previous sections, we focused on understanding how powerlaw spectra influence overfitting and generalization in diffusion models. Now, we shift our focus to understand how and where non-linear models behave differently from linear ones. The purpose of this section is to both discuss the limitations of our approach and to provide a useful starting point for future investigations into non-linear models. We will do this in two steps: first, we will investigate the generalization of nonlinear models beyond the test loss. Second, we directly compare the mappings implemented by trained nonlinear models to linear ones.

**Convergence to a reference model and memorization.** How close to optimal are models trained on a given dataset size $N$? The test loss in non-linear models shows qualitatively similar behavior to

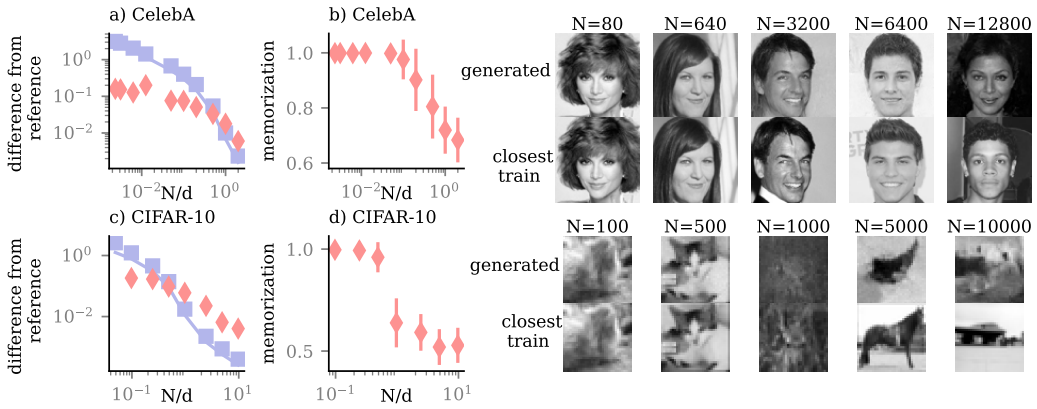



Figure 5: Left column: a) Difference of denoisers from reference model, trained on increasing numbers of data, averaged over 100 test data points. Blue squares are linear models, pink diamonds are U-nets. Blue lines show prediction from replica calculation. b) Similarity between samples generated from U-net architecture diffusion models and closest training data point, averaged over 400 generated data points. c)-d) are equivalent to a)-b) but for CIFAR-10 data. Right column: comparison of generated images vs. closest training example for models trained on increasing number of data.

linear models (see fig. 1), but saturates for both types of models above $N \approx d$. However, we have seen e.g. from the DKL, that model performance still improves there. For general non-gaussian data and non-linear models, however, the DKL is not accessible. Instead, to measure generalization in a more practical way, we compare a model trained with finite $N$ to a reference model trained on the population loss. This measure is readily evaluated across datasets and architectures:

$$\Delta\epsilon_N = \frac{1}{T}\sum_t \left\langle ||\epsilon_N(x^{\text{test}}) - \epsilon_\infty(x^{\text{test}})_t)||^2 \right\rangle_{(x^{\text{test}})} ,$$

where $(x^{\text{test}})$ are noised test samples, $\epsilon_N$ is the mapping obtained from a finite dataset of $N$ samples and $\epsilon_\infty$ is the mapping obtained from an infinite number of samples. $\Delta\epsilon_N$ compares the mappings implemented by a diffusion model optimized on finite $N$ to the best reference. For linear diffusion models, we find

$$\Delta\epsilon_N = L_{\text{test}} - 1 + \frac{1}{T}\sum_t \frac{1-\bar{\alpha}_t}{(1-\bar{\alpha}_t+\gamma_t)} \text{Tr}\left(\text{Id} + \hat{\alpha}_t\Sigma\right)^{-1} . \tag{14}$$

Note that $\Delta\epsilon_N$ and the test loss eq. (4) differ only by terms independent of $N$, which are responsible for the plateau of the test loss. In the unsaturated regime, the test loss is a good proxy for $\Delta\epsilon_N$. For $N > d$, increasing $N$ continues to improve generalization (measured by $\Delta\epsilon_N$), but this effect is no longer reflected in the test loss. In contrast to the test loss, this measure does not saturate above $N \sim d$ and can therefore also measure how non-linear models approach their optima in this regime. In fig. 5 a),c) we show $\Delta\epsilon_N$ for two image datasets, both for linear and non-linear diffusion models, as well as a prediction in the average over draws of the dataset from replica theory (for a detailed calculation, see appendix E.5). In practice, when only finite number of data are available, we choose $\epsilon_\infty$ as the mapping optimized on the largest subset of the data. We find that both for linear and non-linear diffusion models, this difference from the reference model decreases significantly at linear sample complexity.

The most severe consequence of overfitting in diffusion models is *memorization*: when $N$ is small, new samples generated from the trained models can be almost identical to training examples Kadkhodaie et al. [2024], Somepalli et al. [2022]. To measure memorization, we find the closest training image for each generated image based on a detail-based similarity measure (defined in appendix F). The similarity between generated and training set examples is highest at small $N$ and decreases as $N$ increases. We find that in our experiments, memorization diminishes when $N \approx d$ and is related to a lower value of $\Delta\epsilon$. However, the samples generated by linear diffusion models are typically more diverse than those of non-linear diffusion models and do not show qualitatively similar behavior. Nevertheless, recent studies highlight early stopping as an effective measure to prevent memorization Favero et al. [2025b], Bonnaire et al. [2025], which we observe to be beneficial in linear models as well.

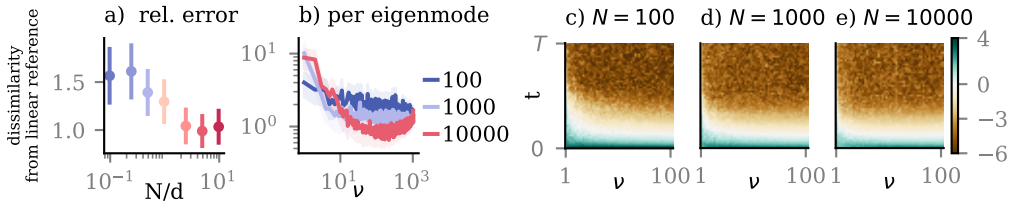



Figure 6: Relative difference of non-linear denoisers from best linear model, per noising step $t$ and direction $\nu$, trained on increasing numbers of data. a) averaged over $\nu$ and $t$, b) averaged over $t$, c) - e) $\log d_{t,\nu}$ per $\nu, t$. All data are averaged over $10^2$ test samples from CIFAR-10 per $t, \nu$.

**Similarity between linear and non-linear models.** Recent experimental studies Wang and Vastola [2023], Li et al. [2024] suggest that diffusion models that have learned more general solutions (that are trained on larger number of data) are more similar to linear diffusion models when $N$ is large. We now test this hypothesis resolved across levels of noise and directions. We define the following distance measure, evaluated in different eigendirections $\nu$ of $\Sigma$:

$$d_{t,\nu} = \frac{\left(\epsilon^N\left(x_t, t\right) - \epsilon^*_\infty\left(x_t, t\right)\right)^2_\nu}{\left|\left(\epsilon^N\left(x_t, t\right) + \eta\right)_\nu \left(\epsilon^*_\infty\left(x_t, t\right) + \eta\right)_\nu\right|} \tag{15}$$

where $\epsilon^N$ is a U-net trained on $N$ examples and $\epsilon^*_\infty$ is a linear model with modest regularization $c = 10^{-2}$, trained on the maximal amount of available data and $\eta = 10^{-3}$ prevents divergences. In fig. 6, we show $d_{t,\nu}$ for the CIFAR-10 dataset, the same is reported for the CelebA dataset in appendix G. Overall, we find that the relative error decays with $N$, $\nu$ and $t$. Indeed for a large extent of $t, \nu$, the difference between linear and non-linear models becomes very small. However, for leading eigenmodes (small $\nu$), the differences between linear and non-linear models grow with $N$. This (small $\nu$, small $t$) is also the regime where we expect the non-Gaussianity of the data to have the largest effect.

**The nullspace effect appears in trained non-linear U-nets.** Finally, we investigate whether trained U-nets underestimate the variability of the data in the nullspace of $\Sigma_0$ similarly to their trained linear counterparts. We expect this effect to be most relevant in the high-resolution regime of image data, where the data dimension is very large and the diffusion model has comparatively few parameters.

To mimic this setting, we train U-Nets of two different sizes on CIFAR-10 data. The larger U-Net has $8.37 \cdot 10^6$ parameters, the smaller U-Net has $2.13 \cdot 10^6$. For the larger U-Net, we find that memorization dominates the sampling behavior (details in appendix H). We now focus on the smaller U-Net.

Concretely, we compare the average variance of generated samples in the nullspace of the training covariance with the corresponding quantity for held-out test data. Let $P^\perp$ be the projector onto the nullspace of $\Sigma_0$, and $u - \mu_0$ a generated sample centered with the empirical mean of the training data. If generated samples remain closer to the empirical training span than genuine test samples, their nullspace variance will be suppressed.

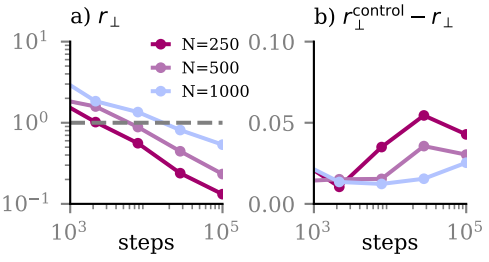



Figure 7: a) $r_\perp$ over training steps for samples generated from U-Net diffusion models vs. variance of test data in the same nullspace, for different number of training data $N$. All models were trained on colored CIFAR-10 data with $d = 3072$. b) Difference from control value, computed from the nullspace of another randomly chosen training set of the same size.

To ensure that this effect is not caused trivially by reproducing individual training examples, we perform the calculation only on generated samples that are not classified as memorized. We then

compute the ratio

$$r_\perp = \frac{\left\langle \left|\left|P^\perp(u-\mu_0)\right|\right|^2 \right\rangle_{\text{non-memorized samples } u}}{\left\langle ||P^\perp(x_0-\mu_0)||^2 \right\rangle_{\text{test data } x_0}} \tag{16}$$

A value of $r_\perp$ below indicates that non-memorized generated samples exhibit less variability than test data in directions absent from the training set.

Out of $4,000$ generates samples, none are classified as memorized training for the model trained on $N = 1000$ training data, only 4 $N = 500$ and 53 at $N = 250$, meaning that memorization plays a minor role for these models. However, the models still underestimate the variance of the data, see fig. 7 a): for all three models, the final values of $r_\perp$ are substantially smaller than one, meaning that also highly non-linear diffusion models underestimate the variance in the nullspace.

Moreover, this underestimation is *specific* to the training set, i.e., the variance in a particular nullspace of the training set is underestimated more severely than that of a randomly chosen subset of the data with the same size. Recomputing $P^\perp_{\text{control}}$ and correspondingly $r_\perp^{\text{control}}$ from such random subsets, we find $r_\perp^{\text{control}} > r_\perp$ consistently for all three models, fig. 7 b).

This shows that the covariance overfitting effect we observe in linear models can persist even in the absence of memorization. It is overfitting on the distributional level: while the model does not reproduce individual training data, it does not generate the statistics of the underlying distribution faithfully.

## 4 Discussion

**Summary.** We identified two regimes for generalization in linear diffusion models, separated by $N = d$. For $N > d$, the test loss saturates, while the Kullback-Leibler divergence continues to decay approximately as $d/N$, independently of the detailed covariance spectrum. This data-independent behavior is consistent with related results for energy-based models Catania et al. [2025], where early stopping and regularization also mitigate overfitting. For $N < d$, overfitting is driven by the low rank of the empirical covariance matrix: $\Sigma_0$ has a nontrivial nullspace, so the training set lacks variability in some directions. Both the test loss and the Kullback-Leibler divergence strongly penalize this missing variability. In this regime, power-law structure is beneficial because faster spectral decay makes the missing directions correspond to smaller population variance.

Regularization introduces an effective cutoff scale $c$, below which covariance structure is no longer resolved. Thus, $c$ selects the relevant scale of variation in the data, echoing connections between principal component analysis, deep learning, and renormalization-group ideas Mehta and Schwab [2014], Bradde and Bialek [2017], Marchand et al. [2023], Kadkhodaie et al. [2023], Lempereur and Mallat [2024]. From a random-matrix perspective, this cutoff is equivalent to ridge/Tikhonov regularization. At finite $N$, the optimal regularization strength decreases with both spectral decay and sample complexity.

**Limitations.** Our analysis is restricted to linear denoisers and, for the averages over training sets, additionally to a Gaussian hypothesis on the data. This makes the generalization dynamics tractable and isolates the role of the covariance spectrum. Despite its simplicity, the linear theory captures important aspects of nonlinear diffusion models, including the qualitative train-test behavior, the dependence on sample complexity, and the agreement in many subleading eigendirections. The comparison with nonlinear U-nets in section 3 also shows where genuinely nonlinear effects enter: the largest discrepancies occur in the leading eigendirections at high signal-to-noise ratio, where non-Gaussian structure and higher-order statistics are expected to matter most.

**Outlook.** An intriguing consequence of our results is that the covariance spectrum affects the severity of overfitting, but not the location of the transition between regimes. This suggests that diffusion models emphasize all directions with finite variability, not only high-variance directions. This contrasts with benign overfitting in supervised learning, where a model may overfit signal while effectively ignoring noise Han et al. [2024]. We showed that this sensitivity is tied to the standard noise-prediction objective: predicting the data instead reweights the test loss toward leading eigendirections, masking missing variability in subleading modes. This provides a concrete mechanism by which the diffusion objective changes which aspects of the data distribution are emphasized, and points to leading eigendirections at high signal-to-noise ratio as the key regime for future theories of how diffusion models learn higher-order statistics.

## Acknowledgements

We are grateful to Alessio Giorlandino for helpful discussions. CM and SG gratefully acknowledge funding from Next Generation EU, in the context of the National Recovery and Resilience Plan, Investment PE1 – Project FAIR "Future Artificial Intelligence Research" (CUP G53C22000440006). SG additionally acknowledges funding from the European Research Council (ERC) for the project "beyond2", ID 101166056, and from the European Union–NextGenerationEU, in the framework of the PRIN Project SELF-MADE (code 2022E3WYTY – CUP G53D23000780001).

## References

Beatrice Achilli, Luca Ambrogioni, Carlo Lucibello, Marc Mézard, and Enrico Ventura. Memorization and Generalization in Generative Diffusion under the Manifold Hypothesis, February 2025. URL http://arxiv.org/abs/2502.09578. arXiv:2502.09578 [cond-mat].

Madhu S Advani, Andrew M Saxe, and Haim Sompolinsky. High-dimensional dynamics of generalization error in neural networks. *Neural Networks*, 132:428–446, 2020.

Giulio Biroli and Marc Mézard. Generative diffusion in very large dimensions. *Journal of Statistical Mechanics: Theory and Experiment*, 2023(9):093402, September 2023. ISSN 1742-5468. doi: 10.1088/1742-5468/acf8ba. URL https://iopscience.iop.org/article/10.1088/1742-5468/acf8ba.

Adam Block, Youssef Mroueh, and Alexander Rakhlin. Generative Modeling with Denoising Auto-Encoders and Langevin Sampling, October 2022. URL http://arxiv.org/abs/2002.00107. arXiv:2002.00107 [stat].

Tony Bonnaire, Raphaël Urfin, Giulio Biroli, and Marc Mézard. Why Diffusion Models Don't Memorize: The Role of Implicit Dynamical Regularization in Training, May 2025. URL http://arxiv.org/abs/2505.17638. arXiv:2505.17638 [cs].

Valentin De Bortoli. Convergence of denoising diffusion models under the manifold hypothesis, May 2023. URL http://arxiv.org/abs/2208.05314. arXiv:2208.05314 [stat].

Serena Bradde and William Bialek. PCA Meets RG. *Journal of Statistical Physics*, 167(3):462–475, May 2017. ISSN 1572-9613. doi: 10.1007/s10955-017-1770-6. URL https://doi.org/10.1007/s10955-017-1770-6.

Francesco Cagnetta, Leonardo Petrini, Umberto M. Tomasini, Alessandro Favero, and Matthieu Wyart. How Deep Neural Networks Learn Compositional Data: The Random Hierarchy Model. *Physical Review X*, 14(3):031001, July 2024. doi: 10.1103/PhysRevX.14.031001. URL https://link.aps.org/doi/10.1103/PhysRevX.14.031001. Publisher: American Physical Society.

Giovanni Catania, Aurelien Decelle, Cyril Furthlehner, and Beatriz Seoane. A theoretical framework for overfitting in energy-based modeling, June 2025. URL http://arxiv.org/abs/2501.19158. arXiv:2501.19158 [cs].

Minshuo Chen, Kaixuan Huang, Tuo Zhao, and Mengdi Wang. Score Approximation, Estimation and Distribution Recovery of Diffusion Models on Low-Dimensional Data. *ArXiv*, abs/2302.07194: null, 2023. doi: 10.48550/arXiv.2302.07194. URL https://www.semanticscholar.org/paper/49ada8f9d765a6ae61ed8e8bbc12c0be37fb2986.

Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion Schrödinger Bridge with Applications to Score-Based Generative Modeling. In *Advances in Neural Information Processing Systems*, volume 34, pages 17695–17709. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/hash/940392f5f32a7ade1cc201767cf83e31-Abstract.html.

Alessandro Favero, Antonio Sclocchi, Francesco Cagnetta, Pascal Frossard, and Matthieu Wyart. How compositional generalization and creativity improve as diffusion models are trained, March 2025a. URL http://arxiv.org/abs/2502.12089. arXiv:2502.12089 [stat].

Alessandro Favero, Antonio Sclocchi, and Matthieu Wyart. Bigger Isn't Always Memorizing: Early Stopping Overparameterized Diffusion Models, September 2025b. URL http://arxiv.org/abs/2505.16959. arXiv:2505.16959 [cs].

Kirsten Fischer, Alexandre René, Christian Keup, Moritz Layer, David Dahmen, and Moritz Helias. Decomposing neural networks as mappings of correlation functions. *Physical Review Research*, 4 (4):043143, November 2022. doi: 10.1103/PhysRevResearch.4.043143. URL https://link.aps.org/doi/10.1103/PhysRevResearch.4.043143. Publisher: American Physical Society.

Anand Jerry George, Rodrigo Veiga, and Nicolas Macris. Analysis of Diffusion Models for Manifold Data, February 2025a. URL http://arxiv.org/abs/2502.04339. arXiv:2502.04339 [math].

Anand Jerry George, Rodrigo Veiga, and Nicolas Macris. Denoising Score Matching with Random Features: Insights on Diffusion Models from Precise Learning Curves, February 2025b. URL http://arxiv.org/abs/2502.00336. arXiv:2502.00336 [cs].

Sebastian Goldt, Marc Mézard, Florent Krzakala, and Lenka Zdeborová. Modeling the Influence of Data Structure on Learning in Neural Networks: The Hidden Manifold Model. *Physical Review X*, 10(4):041044, December 2020. doi: 10.1103/PhysRevX.10.041044. URL https://link.aps.org/doi/10.1103/PhysRevX.10.041044. Publisher: American Physical Society.

Florentin Guth, Zahra Kadkhodaie, and Eero P. Simoncelli. Learning normalized image densities via dual score matching, June 2025. URL http://arxiv.org/abs/2506.05310. arXiv:2506.05310 [cs].

Andi Han, Wei Huang, Yuan Cao, and Difan Zou. On the Feature Learning in Diffusion Models, December 2024. URL http://arxiv.org/abs/2412.01021. arXiv:2412.01021 [stat].

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models, December 2020. URL http://arxiv.org/abs/2006.11239. arXiv:2006.11239 [cs, stat].

Zahra Kadkhodaie, Florentin Guth, Stéphane Mallat, and Eero P. Simoncelli. Learning multi-scale local conditional probability models of images, March 2023. URL http://arxiv.org/abs/2303.02984. arXiv:2303.02984 [cs].

Zahra Kadkhodaie, Florentin Guth, Eero P. Simoncelli, and Stéphane Mallat. Generalization in diffusion models arises from geometry-adaptive harmonic representations, April 2024. URL http://arxiv.org/abs/2310.02557. arXiv:2310.02557 [cs].

Holden Lee, Holden Lee, Jhu Edu, Jianfeng Lu, Yixin Tan, and Yixin Tan. Convergence of score-based generative modeling for general data distributions.

Etienne Lempereur and Stéphane Mallat. Hierarchic Flows to Estimate and Sample High-dimensional Probabilities, May 2024. URL http://arxiv.org/abs/2405.03468. arXiv:2405.03468 [stat].

Xiang Li, Yixiang Dai, and Qing Qu. Understanding Generalizability of Diffusion Models Requires Rethinking the Hidden Gaussian Structure. November 2024. URL https://openreview.net/forum?id=Sk2duBGvrK&referrer=%5Bthe%20profile%20of%20Xiang%20Li%5D(%2Fprofile%3Fid%3D~Xiang_Li44).

Xingchao Liu, Lemeng Wu, Mao Ye, and Qiang Liu. Let us Build Bridges: Understanding and Extending Diffusion Generative Models, August 2022. URL http://arxiv.org/abs/2208.14699. arXiv:2208.14699 [cs].

Tanguy Marchand, Misaki Ozawa, Giulio Biroli, and Stéphane Mallat. Multiscale Data-Driven Energy Estimation and Generation. *Physical Review X*, 13(4):041038, November 2023. ISSN 2160-3308. doi: 10.1103/PhysRevX.13.041038. URL https://link.aps.org/doi/10.1103/PhysRevX.13.041038.

Pankaj Mehta and David J. Schwab. An exact mapping between the Variational Renormalization Group and Deep Learning, October 2014. URL http://arxiv.org/abs/1410.3831. arXiv:1410.3831 [stat].

Jakiw Pidstrigach. Score-Based Generative Models Detect Manifolds, October 2022. URL http://arxiv.org/abs/2206.01018. arXiv:2206.01018 [stat].

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation, May 2015. URL http://arxiv.org/abs/1505.04597. arXiv:1505.04597 [cs].

Jascha Sohl-Dickstein, Eric A Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep Unsupervised Learning using Nonequilibrium Thermodynamics.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion Art or Digital Forgery? Investigating Data Replication in Diffusion Models, December 2022. URL http://arxiv.org/abs/2212.03860. arXiv:2212.03860 [cs].

Yang Song and Stefano Ermon. Generative Modeling by Estimating Gradients of the Data Distribution. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://papers.neurips.cc/paper_files/paper/2019/hash/3001ef257407d5a371a96dcd947c7d93-Abstract.html.

Michele Tumminello, Fabrizio Lillo, and Rosario N. Mantegna. Kullback-Leibler distance as a measure of the information filtered from multivariate data. *Physical Review E*, 76(3):031123, September 2007. doi: 10.1103/PhysRevE.76.031123. URL https://link.aps.org/doi/10.1103/PhysRevE.76.031123. Publisher: American Physical Society.

Binxu Wang and Cengiz Pehlevan. An Analytical Theory of Spectral Bias in the Learning Dynamics of Diffusion Models.

Binxu Wang and John J. Vastola. The Hidden Linear Structure in Score-Based Models and its Application, November 2023. URL http://arxiv.org/abs/2311.10892. arXiv:2311.10892 [cs].

Peng Wang, Huijie Zhang, Zekai Zhang, Siyi Chen, Yi Ma, and Qing Qu. Diffusion Models Learn Low-Dimensional Distributions via Subspace Clustering, December 2024. URL http://arxiv.org/abs/2409.02426. arXiv:2409.02426 [cs].

# A Diffusion models

Diffusion models are designed to reverse the noising process: they implement a mapping $\epsilon_\theta(x_t, t)$ whose parameters $\theta$ are optimized to predict the noise vector $\epsilon_t$. For a dataset $\mathcal{D}$ consisting of $N$ samples in $\mathbb{R}^d$, this amounts to minimizing eq. (3), which we will refer to as the training loss.

Here, we consider the case where for each $t$, the denoiser is implemented by an affine linear mapping

$$\epsilon_\theta(x_t, t) = W_t\left(x_t - \sqrt{\bar{\alpha}_t} b_t\right) \tag{17}$$

where $W_t \in \mathbb{R}^{d\times d}$ is a weight matrix and $b_t \in \mathbb{R}^d$ a bias term. Training the denoiser then amounts to optimizing eq. (3) with respect to $\{W_t, b_t\}_t$. We will also add a standard regularization term $\sum_t \gamma_t \mathrm{Tr} W_t W_t^{\mathrm{T}}$ to the training objective, where the prefactor $\gamma_t$ allows us to apply different levels of regularization to different noising stages $t$. With our choice of architecture and regularization, we find the loss to be

$$L = \frac{1}{dT|\mathcal{D}|}\sum_t \sum_{x_0\in\mathcal{D}} \mathbb{E}_{\epsilon_t}\left\|\epsilon_t - W_t\left(\sqrt{\bar{\alpha}_t}x_t + \sqrt{1-\bar{\alpha}_t}\epsilon_t - \sqrt{\bar{\alpha}_t}b_t\right)\right\|^2 + \gamma_t \mathrm{Tr} W_t W_t^{\mathrm{T}} . \tag{18}$$

## A.1 Optimizing the loss

Since $W_t$, $b_t$ are not coupled in eq. (3) across different values of $t$, we can find their optima independently for each noising stage. Inserting eq. (17) into eq. (3), we find that the contribution for each $t$ decomposes into a data-dependent term and one which depends only on the additive noise

$$\mathbb{E}_{\epsilon_t}\left\|\epsilon_t - \epsilon_\theta(x_t, t)\right\|^2 = \bar{\alpha}_t \left\|W_t(x_0 - b_t)\right\|^2 + \mathbb{E}_{\epsilon_t}\left\|\left(1 - \sqrt{1-\bar{\alpha}_t}W_t\right)\epsilon_t\right\|^2 . \tag{19}$$

We will not consider variations which arise due to the fact that only a finite number of noised samples are used at each training step, hence in eq. (19) we take the average over infinitely many realizations of $\epsilon_t$.

Eq. (19) shows that the loss contains only terms either linear or quadratic in the training data $x_0$. Consequently, only the first two moments of the data, or, equivalently, the empirical mean $\mu_0$ and covariance $\Sigma_0$

$$\mu_0 := \frac{1}{N}\sum_{x_0\in\mathcal{D}} x_0 , \qquad \Sigma_0 := \frac{1}{N}\sum_{x_0\in\mathcal{D}} (x_0-\mu_0)(x_0-\mu_0)^{\mathrm{T}} =: \sum_\nu \lambda_\nu^0 e_\nu^0 \left(e_\nu^0\right)^T \tag{20}$$

of the training data. With this definitions, we now use that eq. (18) is quadratic in $W_t, b_t$, hence it is convex with the unique minimum

$$b_t^* = \mu_0 , \qquad W_t^* = \frac{\sqrt{1-\bar{\alpha}_t}}{\bar{\alpha}_t\Sigma_0 + (1-\bar{\alpha}_t+\gamma_t)\,\mathrm{Id}} . \tag{21}$$

At the optimum, $W_t^*, b_t^*$, we find the irreducible, or residual, loss

$$R = \sum_t\sum_\nu \left[\frac{\bar{\alpha}_t\lambda_\nu^0 + \gamma_t}{(\bar{\alpha}_t\lambda_\nu^0 + (1-\bar{\alpha}_t+\gamma_t))}\right] . \tag{22}$$

which is the minimal value of the training loss which a linear denoiser can achieve.

Once trained, the denoiser is then used to iteratively generate new samples, we provide a description of the generation process in the next section.

## A.2 The generation process

The generation process follows the reverse direction to the noising process, starting from pure white noise $u_0 \sim \mathcal{N}(0, \mathrm{Id})$ and culminating in a new sample $u_T$. For sampling, we will use $s = T - t$ as an iteration index and $u$ as a dynamical variable to distinguish the noising process from the denoising process. For sampling, we will use $s = T - t$ as an iteration index and $u$ as a dynamical variable to distinguish the noising process from the denoising process. The iteration reads

$$u_{s+1} = \mu_\theta(u_s, T-s) + \sigma_{T-s}\xi, \qquad \xi \sim \mathcal{N}(0, \mathrm{Id}) . \tag{23}$$

where we defined

$$\mu_\theta(x_t, t) := \frac{x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(x_t, t)}{\sqrt{\alpha_t}} + \sqrt{(1-\sigma_t^2) - \bar{\alpha}_{t-1}}\epsilon_\theta(x_t, t) . \tag{24}$$

These two equations can be understood as first predicting $x_0 \approx \frac{x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta}{\sqrt{\bar{\alpha}_t}}$, then adding back "noise" in the form of $\sigma_t\xi + \sqrt{(1-\sigma_t^2) - \bar{\alpha}_{t-1}}\epsilon_\theta$.

## B Sampling dynamics of affine linear denoisers

Affine linear denoisers sample from Gaussian distributions: Starting from an initial Gaussian random variable $u_0 \sim \mathcal{N}(0, \mathrm{Id})$, $u_T$ is a linear map of Gaussian random variables and constants, hence it is also Gaussian. Up to orders of $T^{-1}$, affine linear networks reproduce the mean $\mu_0$ and covariance $\Sigma_0$ of the training data. In this section we compute how the mean and covariance of the samples evolve under the the iterative denoising process specified in eq. (23). Before we do so, we note that to sample from a given Gaussian distribution with mean $\mu_0$ and covariance $\Sigma_0$, no iterative process is necessary. Rather, given $u_0 \sim \mathcal{N}(0, \mathrm{Id})$, we can generate a sample $u$ with the aforementioned statistics with a simple linear transform

$$u_T = \sqrt{\Sigma_0} u_0 + \mu_0 \,. \tag{25}$$

we will see in the following that the iterative sampling process approaches the same statistics of $u_T$. In this manuscript, we do not consider the fact that the noising process is discrete, rather we will treat sampling time as continuous. However, we will highlight corrections which arise due to $\bar{\alpha}(0) \neq 1$ and $\bar{\alpha}(T) \neq 0$.

### B.1 Continuous time limit

For the following calculations, it will be useful to write a continuous time version of the denoising process. To this end, we also consider a rescaled time in which the time arguments in the sampling process are incremented by $h = T^{-1}$, rather than increments of 1 as in eq. (23), thus both denoising time $s$ and noising time $t$ run from zero to one with $t = 1 - s$. We now assume $\beta(t) = \mathcal{O}(h), \forall t$. This is reasonable, since we expect the changes in every step of the diffusion process to be small. We now define

$$\hat{\beta}(t) := \frac{\beta_t}{h} \qquad\qquad \hat{\sigma}(t) := \frac{\sigma_t}{\sqrt{h}} \,. \tag{26}$$

For $\bar{\alpha}(t)$,we use that $\bar{\alpha}(t+h) = \left(1 - \hat{\beta}(t)h\right) \bar{\alpha}(t)$, so in the limit $T \to \infty$, or equivalently $h \to 0$,

$$\frac{d\bar{\alpha}(t)}{dt} = -\hat{\beta}(t)\bar{\alpha}(t) \,. \tag{27}$$

This differential equation admits a formal solution, namely

$$\bar{\alpha}(t) = e^{-\zeta(t)}, \zeta(t) := \int_0^t ds\, \hat{\beta}(s) \,. \tag{28}$$

### B.2 Fokker-Planck equation

With the definition of the continuous noising/denoising time limit in hand, we now write eq. (23) as a linear stochastic differential equation. We find

$$du(s) = (m(s)u(s) + c(s))\, ds + \hat{\sigma}(1-s) dZ_s$$

where $m_s, c_s$ are found by inserting eq. (24) and eq. (21) into eq. (23) and $Z_s$ is a Wiener process. We find

$$\begin{aligned} m(s) &= \frac{\hat{\beta}(1-s)}{2}\mathrm{Id} - \frac{1}{2}\frac{\left(\hat{\sigma}^2(1-s) + \hat{\beta}(1-s)\right)}{\bar{\alpha}(1-s)\Sigma_0 + (1-\bar{\alpha}(1-s))\,\mathrm{Id}}, \\ c(s) &= \frac{1}{2}\left(\hat{\sigma}^2(1-s) + \hat{\beta}(1-s)\right)\frac{\sqrt{\bar{\alpha}(1-s)}}{\bar{\alpha}(1-s)\Sigma_0 + (1-\bar{\alpha}(1-s))\,\mathrm{Id}}\mu_0 \,. \end{aligned}$$

Observe that $m(s)$ is diagonal in the eigenbasis of $\Sigma_0$. Moving into this basis, we can now solve for the statistics of the sampling process in a decoupled manner, as in this basis, all entries of $u(s)$ are statistically independent. In the following calculation, we will keep one direction $u_\nu$ fixed, dropping the index $\nu$ fo brevity. We use the Fokker-Planck equation to write down the differential equation for the density $\rho(u,t)$ which describes this variable. We have

$$\partial_s \rho(u,s) = -\partial_u \left[(m(s)u(s) + c(s))\,\rho(u,s)\right] + \frac{1}{2}\partial_u^2 \left[\hat{\sigma}(1-s)^2 \rho(u,s)\right] \,. \tag{29}$$

Since we know that this is a Gaussian process, we make the following Ansatz for the density

$$\rho(u,s) = \frac{1}{\sqrt{2\pi\sigma_u(s)^2}} \exp\left[-\frac{(u(s)-\mu_u(s))^2}{2\sigma_u(s)^2}\right]$$

defining $\mu_u(s), \sigma_u(s)^2$ as the mean and variance of the samples at sampling time $s$, respectively. With this Ansatz we find that

$$\partial_s(\sigma_u(s)^2) = 2m(s)\sigma_u(s)^2 + \hat{\sigma}(1-s)^2 \qquad \partial_s\mu_u(s) = m(s)\mu_u(s) + c(s). \tag{30}$$

which admit the solutions

$$\mu_u(s) = \exp\left(\int_0^s dv\, m(v)\right)\left[\int_0^s dv\, \exp\left(-\int_0^v dw\, m(w)\right) c(v) + \mu_u(0)\right] \tag{31}$$

$$\sigma_u(s)^2 = \exp\left(2\int_0^s dv\, m(v)\right)\left[\int_0^s dv\, \exp\left(-2\int_0^v dw\, m(w)\right) \hat{\sigma}(1-v)^2 + \sigma_u(0)^2\right] \tag{32}$$

The initial conditions are $\sigma_u(0) = 1$, $\mu_u(0) = 0$ since $u(0) \sim \mathcal{N}(0, \mathrm{Id})$. We will find closed form solutions for these integrals for two choices of $\hat{\sigma}$ in the next step.

### B.3 Solutions for mean and covariance of samples

We will first simplify some of the integrals to solve these two equations. To find the variance $\sigma_u(s)^2$, we first solve

$$\int_0^s dv\hat{\beta}(1-v) = \ln\left(\frac{\bar{\alpha}(1-s)}{\bar{\alpha}(1)}\right)$$

$$\int_0^s dv\, \frac{-\hat{\beta}(1-v)}{\bar{\alpha}(1-v)(\lambda^0-1)+1} = -\ln\left(\frac{\bar{\alpha}(1-s)}{\bar{\alpha}(1)}\right) + \ln\left(\frac{\bar{\alpha}(1-s)(\lambda^0-1)+1}{\bar{\alpha}(1)(\lambda^0-1)+1}\right)$$

Second, we have $c(s) = \sqrt{\bar{\alpha}(1-s)}\left(m(s) - \frac{\hat{\beta}(1-s)}{2}\right)\mu_0$. With

$$\begin{aligned}&\int_0^s dv\, \exp\left(-\int_0^v dw\, m(w)\right)\sqrt{\bar{\alpha}(1-v)}\left(m(v) - \frac{\hat{\beta}(1-v)}{2}\right)\\ =&\sqrt{\bar{\alpha}(1-s)}\, \exp\left(-\int_0^s dw\, m(w)\right) - \sqrt{\bar{\alpha}(1)}\end{aligned}$$

we then find

$$\mu_u(s) = \sqrt{\bar{\alpha}(1-s)}\mu_0 - \exp\left(\int_0^s dv\, m(v)\right)\sqrt{\bar{\alpha}(1)}\mu_0$$

We now treat two different scenarios, $\hat{\sigma}^2(t) = \hat{\beta}(t)$ and $\hat{\sigma}(t) = 0$.

#### B.3.1 $\hat{\sigma}(t) = 0$

In this case, we have $\int_0^s dv\, m(v) = \frac{1}{2}\ln\left(\frac{\bar{\alpha}(1-s)(\lambda^0-1)+1}{\bar{\alpha}(1)(\lambda^0-1)+1}\right)$, hence

$$\begin{aligned}\mu_u(s) &= \sqrt{\bar{\alpha}(1-s)}\mu_0 - \sqrt{\bar{\alpha}(1)\frac{\bar{\alpha}(1-s)(\lambda^0-1)+1}{\bar{\alpha}(1)(\lambda^0-1)+1}}\mu_0\\ \sigma_u(s)^2 &= \frac{\bar{\alpha}(1-s)(\lambda^0-1)+1}{\bar{\alpha}(1)(\lambda^0-1)+1}\end{aligned} \tag{33}$$

Since $\bar{\alpha}(1)$ vanishes exponentially with $\zeta$, and $\bar{\alpha}(0) = 1 + \mathcal{O}(h)$, we find that for a long noising trajectory of many steps, $\Sigma_u = \Sigma_0 + \mathcal{O}(h)$ and $\mu_u = \mu_0 + \mathcal{O}(h)$, reproduce the empirical mean and covariance of the training set to good approximation.

### B.3.2 $\hat{\sigma}^2(t) = \hat{\beta}(t)$

In this case $\int_0^s dv\, m(v) = \ln\left(\frac{\bar{\alpha}(1-s)(\lambda^0-1)+1}{\bar{\alpha}(1)(\lambda^0-1)+1}\right) + \frac{1}{2}\ln\left(\frac{\bar{\alpha}(1)}{\bar{\alpha}(1-s)}\right)$, hence we find the sampling mean to be

$$\mu_u(s) = \sqrt{\bar{\alpha}(1-s)}\mu_0 - \sqrt{\frac{\bar{\alpha}(1)^2}{\bar{\alpha}(1-s)}\frac{\bar{\alpha}(1-s)(\lambda^0-1)+1}{\bar{\alpha}(1)(\lambda^0-1)+1}}\mu_0\,.$$

To evaluate eq. (32) for the covariance we first solve the following integral

$$\int_0^s dv\, \exp\left(-2\int_0^v dw\, m(w)\right)\hat{\beta}(1-v) = \frac{\left(\bar{\alpha}(1)(\lambda^0-1)+1\right)^2}{\bar{\alpha}(1)} \cdot \left(\frac{\bar{\alpha}(1-s)}{\bar{\alpha}(1-s)(\lambda^0-1)+1} - \frac{\bar{\alpha}(1)}{\bar{\alpha}(1)(\lambda^0-1)+1}\right).$$

Inserting this into eq. (32), we find for the sampling covariance

$$\sigma_u(s)^2 = \bar{\alpha}(1-s)(\lambda^0-1)+1 + \frac{\bar{\alpha}(1)^2(\lambda^0-1)}{\bar{\alpha}(1-s)}\left(\frac{\bar{\alpha}(1-s)(\lambda^0-1)+1}{\bar{\alpha}(1)(\lambda^0-1)+1}\right)^2$$

Again, since $\bar{\alpha}(1)$ vanishes exponentially with $\zeta$, and $\bar{\alpha}(0) = 1 + \mathcal{O}(h)$, we find that for a long noising trajectory of many steps, $\Sigma_u(1) = \Sigma_0 + \mathcal{O}(h)$ and $\mu_u(1) = \mu_0 + \mathcal{O}(h)$, meaning that the sampling mean and covariance reproduce the empirical mean and covariance of the training set to good approximation.

In the case of finite regularization $\gamma_t = c\bar{\alpha}_t$, we must replace $\lambda^0$ with $\lambda^0 + c$ in all formulae. This shows that both regularization $\bar{\alpha}(0) \neq 1$ bias the sampler towards a covariance matrix with an additional, spherical term.

## C Learning dynamics of linear denoisers

Throughout this section, we will assume that all data sets are centered, hence that $\mu_0 = \mu = 0$ and that the bias terms $b_t$ are initialized at zero, corresponding to their optimal value in this case. We introduce a training time $\tau$ and a learning rate $\eta$. At each training step, we will update the parameters of the linear network $\theta$ according to

$$\theta(\tau + d\tau) - \theta(\tau)d\tau = -\eta\nabla_\theta L$$

We will treat the dynamics of $W_t$ it in the eigenbasis of the empirical covariance matrix,

$$W_t(\tau) = \sum_\nu w_{\mu,\nu,t}(\tau)\, e^0_\mu \otimes e^0_\nu$$

Inserting this expression into the training loss, we find

$$L = \frac{1}{dT}\sum_t \sum_{\mu,\nu}\left[\left(\bar{\alpha}_t\lambda^0_\nu + \gamma_t + 1 - \bar{\alpha}_t\right) w^2_{\mu,\nu,t} - 2\sqrt{1-\bar{\alpha}_t}w_{\mu,\nu,t}\right] + d$$

This expression shows that all entries in $w_{\mu,\nu,t}(\tau)$ decouple and we can treat the evolution of the weight matrices elementwise. Taking the derivative and using the definition of $W^*_T$ (eq. (21)) we find, that in the limit $d\tau \to 0$

$$w_{\mu,\nu,t}(\tau) = w^*_{\mu,\nu,t} + \exp\left\{-2\frac{\eta}{dT}\left[\bar{\alpha}_t\lambda^0_\nu + (1-\bar{\alpha}_t+\gamma_t)\right]\tau\right\}\left(w_{\mu,\nu,t}(0) - w^*_{\mu,\nu,t}\right). \tag{34}$$

This expression shows that through training, the entries of $W_t$ approach their optimal value exponentially with rate $2\frac{\eta}{dT}\left[\bar{\alpha}_t\lambda^0_\nu + (1-\bar{\alpha}_t+\gamma_t)\right]$ corresponding to different spatial directions. The rate $\bar{\alpha}_t\lambda^0_\nu + 1 - \bar{\alpha}_t + \gamma_t$ corresponds precisely to the denominator of the terms in eq. (4), whose minimal values lead to the most severe overfitting.

## D Predicting the data instead of the noise

In this appendix, we detail the calculations corresponding to the objective eq. (12). In principle, both objectives are equivalent, since, if one has a model $\epsilon_\theta(x_t, t)$ that predicts the noise, one can use it to define a function $f_\theta(x_t, t)$ to predict the data, using

$$x_t \hat{=} \sqrt{\bar{\alpha}_t} f^*_\theta(x_t, t) + \sqrt{1-\bar{\alpha}_t}\epsilon^*_\theta(x_t, t)\,. \tag{35}$$

Again, we parametrize the network as an affine linear function

$$f_\theta(x_t, t) = V_t x_t + r_t$$

where $V_t$ is a matrix and $r_t$ a vector, both are learnable parameters of the model. Then one finds that the minimum of eq. (12) is reached when

$$r^*_t = -\left(\sqrt{\bar{\alpha}_t} V^*_t - \mathrm{Id}\right)\mu_0 \qquad V^*_{t,\mu\nu} = \frac{\sqrt{\bar{\alpha}_t}\Sigma^0}{\bar{\alpha}_t\Sigma^0 + (1-\bar{\alpha}_t)\,\mathrm{Id}}\,,$$

which is consistent with eq. (35). Inserting this into the training loss, we find

$$R = \frac{1}{dT}\sum_t \sum_\nu \left(\lambda^0_\nu - \frac{\bar{\alpha}_t \left(\lambda^0_\nu\right)^2}{\bar{\alpha}_t\lambda^0_\nu + 1 - \bar{\alpha}_t}\right)\,,$$

which is the irreducible error/ residual loss. The test loss, instead, becomes

$$L_{\text{test},D} = \frac{1}{dT}\sum_t\sum_\nu\left(\Sigma_{\nu\nu} - \bar{\alpha}_t\left(\lambda^0_\nu\right)^2\left(\frac{\bar{\alpha}_t\Sigma_{\nu\nu} + (1-\bar{\alpha}_t)}{\left(\bar{\alpha}_t\lambda^0_\nu + 1 - \bar{\alpha}_t\right)^2}\right) - 2\bar{\alpha}_t\lambda^0_\nu\frac{(1-\bar{\alpha}_t)\left(\Sigma_{\nu\nu} - \lambda^0_\nu\right)}{\left(\bar{\alpha}_t\lambda^0_\nu + (1-\bar{\alpha}_t)\right)^2}\right) \tag{36}$$

For the learning dynamics, assuming $r_t = \mu_0 = 0$ again, one finds

$$\begin{aligned}\partial_\tau V_{t,\mu\nu}(\tau) &= -\frac{2\eta}{dT}\left(\bar{\alpha}_t\lambda^0_\nu + 1 - \bar{\alpha}_t\right)\left(V_{t,\mu\nu} - V^*_{t,\mu\nu}\right)\\ \Rightarrow V_{t,\mu\nu}(\tau) &= V^*_{t,\mu\nu} + \exp\left(-\eta_{t,\nu}\tau\right)\left(V_{t,\mu\nu}\left(0\right) - V^*_{t,\mu\nu}\right)\end{aligned}$$

This means that the individual modes of $V_t$ are learned at the same speed as the corresponding modes of $W_t$.

## E Replica theory for linear denoisers at finite $N$

In this appendix, we derive summary statistics for linear denoisers optimized using the empirical covariance matrix $\Sigma_0$ for different sample sizes $N$. We will assume that the training data originates from a centered Gaussian $\rho \sim \mathcal{N}(0, \Sigma)$, where we define the "true" covariance matrix $\Sigma$ to be

$$\Sigma = R\Lambda R^{\mathrm{T}}, \quad \Lambda = \begin{pmatrix}\lambda_1 & \dots & 0\\ \vdots & \ddots & \vdots\\ 0 & \dots & \lambda_d\end{pmatrix} \tag{37}$$

with $R$ a fixed rotation matrix, therefore $|R| = 1, R^{\mathrm{T}}R = \mathrm{Id}$. We parametrize the empirical covariance matrix $\mathrm{Id} + \hat{\alpha}_t\Sigma_0$ in the following way

$$\Sigma_0 = \frac{1}{N}\sum_{\beta=1}^N \Sigma^{\frac{1}{2}}x^\beta\left(\Sigma^{\frac{1}{2}}x^\beta\right)^{\mathrm{T}}, \quad x^\beta \sim \mathcal{N}(0, \mathrm{Id})\;\forall\beta = 1, \dots, N$$

We are now interested in statistics of the inverse of the related random matrix $\mathrm{Id} + \hat{\alpha}\Sigma_0$. We define the following quantities

$$\begin{aligned}f_g(J) =& \frac{1}{d}\int\prod_\beta d\rho\left(x^\beta\right)\ln Z(J, \Sigma_0)\\ Z(J, \Sigma_0) :=& \left|\mathrm{Id} + \hat{\alpha}\Sigma_0 + RJg(\Lambda)R^{\mathrm{T}}\right|^{-\frac{1}{2}}\\ =& \int\frac{d\eta}{\sqrt{2\pi}^d}\exp\left(-\frac{1}{2}\eta^{\mathrm{T}}\left[\mathrm{Id} + \frac{\hat{\alpha}}{N}\sum_{\beta=1}^N\Sigma^{\frac{1}{2}}x^\beta\left(\Sigma^{\frac{1}{2}}x^\beta\right)^{\mathrm{T}} + RJg(\Lambda)R^{\mathrm{T}}\right]\eta\right)\end{aligned} \tag{38}$$

The function $f$ then plays the role of a generating functional for the moments of the inverse of $\mathrm{Id} + \hat{\alpha}_t \Sigma_0$, e.g.

$$\frac{1}{d}\left\langle (\mathrm{Id} + \hat{\alpha}_t \Sigma_0)^{-1} \right\rangle_{\Sigma_0} = R\left(-2\frac{d}{dJ} f(J)|_{J=0, g(\Lambda)=\mathrm{Id}}\right) R^{\mathrm{T}}$$

For the relevant quantities computed in this manuscript, it will be sufficient to compute $f$ for diagonal $J$, $J_{ij} := \delta_{ij} J_i$. This is because all quantities can be written as traces of matrix products, and choosing $J$ thus corresponds to choosing the basis in which we evaluate the trace to be given by $R$.

The difficulty in computing $f$ then arises from the fact that all the integrals in $x^\beta$ are coupled in the logarithm. To evaluate the integral, we now use the replica trick, which consists of re-writing the logarithm as

$$\langle \ln Z(J, \Sigma_0) \rangle_{\Sigma_0} = \lim_{n \to 0} \frac{1}{n} \left( \langle Z(J, \Sigma_0) \rangle_{\Sigma_0}^n - 1 \right) \tag{39}$$

The replica trick then consists of evaluating $\langle Z(J, \Sigma_0) \rangle_{\Sigma_0}^n$ for integer values of $n$ and then taking the limit $n \to 0$. To compute $\langle Z(J, \Sigma_0) \rangle_{\Sigma_0}^n$, we first write the power as an integral over $n$ independent variables $\eta^\alpha$, where $\alpha$ is the replica index,

$$\langle Z(J, \Sigma_0) \rangle_{\Sigma_0}^n = \int \left( \prod_{\alpha=1}^n \frac{d\eta^\alpha}{\sqrt{2\pi}^d} \right) \left\langle \exp\left( \sum_{\alpha=1}^n -\frac{1}{2} \left(\eta^\alpha\right)^{\mathrm{T}} \left(\mathrm{Id} + \hat{\alpha}_t \Sigma_0 + R J R^{\mathrm{T}} f(\Sigma)\right) \eta^\alpha \right) \right\rangle_{\Sigma_0} \tag{40}$$

In the following section, we will simplify this expression via a change of variables to a set of summary statistics.

### E.1 Introducing auxiliary variables

We first isolate the terms depending on $x^\beta$ from the expression.

$$\begin{aligned} \langle Z \rangle_{\Sigma_0}^n = \int \left( \prod_{\alpha=1}^n \frac{d\eta^\alpha}{\sqrt{2\pi}^d} \right) \exp\left( \sum_{\alpha=1}^n \left[ -\frac{1}{2} \left(\eta^\alpha\right)^{\mathrm{T}} \left(\mathrm{Id} + R J f(\Lambda) R^{\mathrm{T}}\right) \right] \right) \\ \cdot \int \left( \prod_{\beta=1}^N \frac{dx^\beta}{\sqrt{2\pi}^d} \right) \exp\left( -\frac{1}{2} \sum_\beta \left(x^\beta\right)^{\mathrm{T}} \left( \frac{\hat{\alpha}}{N} \sum_\alpha \Sigma^{\frac{1}{2}} \eta^\alpha \left(\Sigma^{\frac{1}{2}} \eta^\alpha\right)^{\mathrm{T}} + \mathrm{Id} \right) x^\beta \right) \end{aligned} \tag{41}$$

To simplify the expression a bit, we now change variables to $\mu^\alpha = \Sigma^{\frac{1}{2}} \eta^\alpha$, we then find that we can isolate one factor in which $\Sigma$ appears, but not the samples $x^\beta$, and vice versa. Additionally, note that for given $\mu^\alpha$, the second line is just the $N-$th power of the first line. Both factors are coupled together by the fact that $\mu^\alpha$ appear in both factors. We thus simplify to

$$\begin{aligned} \langle Z \rangle_{\Sigma_0}^n = \int \left( \prod_{\alpha=1}^n \frac{d\mu^\alpha}{\sqrt{2\pi}^d} \right) \exp\left( -\frac{1}{2} \sum_{\alpha=1}^n \left(R^{\mathrm{T}} \mu^\alpha\right)^{\mathrm{T}} \Lambda^{-\frac{1}{2}} \left(\mathrm{Id} + J g(\Lambda)\right) \Lambda^{-\frac{1}{2}} \left(R^{\mathrm{T}} \mu^\alpha\right) - \frac{n}{2} \ln |\Lambda| \right) \\ \cdot \left[ \underbrace{\int \frac{dx}{\sqrt{2\pi}^d} \exp\left( -\frac{1}{2} x^{\mathrm{T}} \left( \frac{\hat{\alpha}}{N} \sum_\alpha \mu^\alpha \left(\mu^\alpha\right)^{\mathrm{T}} + \mathrm{Id} \right) x \right)}_{=:G} \right]^N \end{aligned}$$

Another rotation of both $x^\beta$ and $\mu^\alpha$ by $R^{\mathrm{T}}$ then eliminates $R$ from the expression, leaving all other terms unchanged. We now simplify the latter integral, $G$. Our goal is to have all directions $i$ of $\mu^\alpha$ decouple. To this end, we define our first auxiliary variable

$$R_\alpha = \frac{1}{\sqrt{d}} x^{\mathrm{T}} \mu^\alpha \tag{42}$$

and enforce this definition with a Dirac delta in the integral, using that $\delta(r-m) = \frac{1}{2\pi}\int d\tilde{r}\exp\left(i\tilde{r}\left[r-m\right]\right)$. This yields

$$G = \int \frac{dx}{\sqrt{2\pi}^d}\prod_\alpha \frac{dR_\alpha d\tilde{R}_\alpha}{2\pi}\exp\left(-\frac{1}{2}\frac{d\hat{\alpha}}{N}\sum_\alpha R_\alpha^2 - \frac{x^2}{2} + i\tilde{R}_\alpha\left(R_\alpha - \frac{1}{\sqrt{d}}x^{\mathrm{T}}\mu^\alpha\right)\right)$$
$$= \int\prod_\alpha dR_\alpha\frac{d\tilde{R}_\alpha}{2\pi}\exp\left(-\frac{1}{2}\frac{d\hat{\alpha}}{N}\sum_\alpha R_\alpha^2 + i\tilde{R}_\alpha R_\alpha\right)\int\frac{dx}{\sqrt{2\pi}^d}\exp\left(-\frac{x^2}{2} - i\frac{1}{\sqrt{d}}\tilde{R}_\alpha x^{\mathrm{T}}\mu^\alpha\right)$$

We see that the integral over $x$ is a Gaussian integral which can be solved exactly, yielding

$$G = \int\prod_\alpha dR_\alpha\frac{d\tilde{R}_\alpha}{2\pi}\exp\left(\sum_\alpha\left(-\frac{1}{2}\frac{d\hat{\alpha}}{N}R_\alpha^2 + i\tilde{R}_\alpha R_\alpha\right) - \sum_{\alpha_1,\alpha_2}\frac{\tilde{R}_{\alpha_1}\tilde{R}_{\alpha_2}}{2d}\left(\mu^{\alpha_1}\right)^{\mathrm{T}}\mu^{\alpha_2}\right)$$

Importantly, this quantity depends only on the replica overlaps $\left(\mu^{\alpha_1}\right)^{\mathrm{T}}\mu^{\alpha_2}$, which brings us to our second auxiliary variable:

$$Q_{\alpha_1,\alpha_2} := \frac{1}{d}\left(\mu^{\alpha_1}\right)^{\mathrm{T}}\mu^{\alpha_2} \tag{43}$$

Using the Hubbard-Strantonivic transform backwards to also eliminate the integrals over all $R_\alpha$, we find

$$G = \int\sqrt{\frac{N}{d\hat{\alpha}}}^n\int\prod_\alpha\frac{d\tilde{R}_\alpha}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}\tilde{R}^{\mathrm{T}}\left(\frac{Q}{d}+\mathrm{Id}\frac{N}{d\hat{\alpha}}\right)R\right) = \sqrt{\frac{N}{d\hat{\alpha}}}^n\left|Q+\mathrm{Id}\frac{N}{d\hat{\alpha}}\right|^{-\frac{1}{2}}$$

Inserting the result for $G$ into the expression as well as enforcing the definition of $Q$ with another Dirac delta, we find

$$\langle Z\rangle_{\Sigma_0}^n = \prod_i d\rho_i(\lambda_i)\left(\prod_{\alpha_1=1}^n d\mu^{\alpha_1}\prod_{\alpha_2=1}^{\alpha_1}dQ_{\alpha_1,\alpha_2}\frac{d\tilde{Q}_{\alpha_1,\alpha_2}}{2\pi}\right)\exp\left(S\left(\lambda,\{\mu_\alpha\}_\alpha,Q,\tilde{Q}\right)\right)$$

$$\begin{aligned}S\left(\lambda,\{\mu_\alpha\}_\alpha,P,\tilde{P}\right) =& -\frac{1}{2}\sum_i\left[\lambda_i^{-1}\left(1+J_i g(\lambda_i)\right)\sum_\alpha\left(\mu_i^\alpha\right)^2\right]\\ &+ i\sum_{\alpha_1\leq\alpha_2}\tilde{Q}_{\alpha_1,\alpha_2}\left(Q_{\alpha_1,\alpha_2}-\frac{1}{d}\left(\mu^{\alpha_1}\right)^{\mathrm{T}}\mu^{\alpha_2}\right)\\ &+ N\ln G(Q) - \frac{n}{2}\ln|\lambda_i|\end{aligned}$$

Here we have also explicitly used the fact that we chose $J$ to be diagonal in the eigenspace of $\Sigma$. We now also solve the integral over $\mu^\alpha$ by exploiting that all spatial directions in the expression are decoupled. The $\mu^\alpha$ dependent part of the integral is then given by

$$S_i(\tilde{Q}) = \ln\int\prod_\alpha\frac{d\mu_i^\alpha}{\sqrt{2\pi}}\exp\left(-\frac{1}{2\lambda_i}\left(1+J_i f(\lambda)\right)\sum_\alpha\left(\mu_i^\alpha\right)^2 - \frac{n}{2}\ln|\lambda_i| - \frac{i}{d}\sum_{\alpha_1\leq\alpha_2}\tilde{Q}_{\alpha_1,\alpha_2}\mu_i^{\alpha_1}\mu_i^{\alpha_2}\right)$$
$$= \ln\sqrt{\lambda_i}^{-n}\left|\lambda_i^{-1}\mathrm{Id}\left(1+J_i g(\lambda_i)\right)+\frac{i}{d}\,\mathrm{diag}\tilde{Q}+\frac{i}{d}\tilde{Q}\right|^{-\frac{1}{2}}$$

With this, we find that the only remaining integrals are in $Q$ and $\tilde{Q}$. Assuming that $N, \tilde{Q} = \mathcal{O}(d)$, we now pull out a factor $d$

$$\bar{Z}^n(j) = \int\prod_{\alpha_1\leq\alpha_2}^n\frac{dQ_{\alpha_1,\alpha_2}d\tilde{Q}_{\alpha_1,\alpha_2}}{2\pi}\exp\left(dS\left(Q,\tilde{Q}\right)\right)$$
$$S\left(Q,\tilde{Q}\right) = \left[i\sum_{\alpha_1\leq\alpha_2}\frac{1}{d}\tilde{Q}_{\alpha_1,\alpha_2}Q_{\alpha_1,\alpha_2} + \frac{1}{d}\sum_i S_i(\tilde{Q}) + \frac{N}{d}\ln G(Q)\right] \tag{44}$$

We will not solve these integrals explicitly. Rather, we will approximate the integral by $\exp\left(dS\left(Q^*,\tilde{Q}^*\right)\right)$, where $Q^*, \tilde{Q}^*$ are the maxima of $S$. This is because due to the common prefactor $d$, the integral is assumed to concentrate around a single point for $d\to\infty$, the saddle point.

### E.2 Saddle point approximation for any $n$

Before we find $Q^*, \tilde{Q}^*$, we introduce simplification in the form of a replica symmetric Ansatz

$$Q_{\alpha_1,\alpha_2} = q\delta_{\alpha_1,\alpha_2} + p(1-\delta_{\alpha_1,\alpha_2}) \tag{45}$$

$$\frac{i}{d}\tilde{Q} = \tilde{q}\delta_{\alpha_1,\alpha_2} + \tilde{p}(1-\delta_{\alpha_1,\alpha_2})\,. \tag{46}$$

Parameterizing $Q, \tilde{Q}$ in this way implicitly assumes all replicas are equivalent. With this simplification, we can explicitly diagonalize $Q = npe_1e_1^{\mathrm{T}} + (q-p)\mathrm{Id}$ with $\forall i = 1,\dots,n$, where $e_1^\alpha = \frac{1}{\sqrt{n}}\forall\alpha = 1,\dots n$. Likewise we find $\frac{i}{d}\,\mathrm{diag}\tilde{Q} + \frac{i}{d}\tilde{Q} = n\tilde{p}e_1e_1^{\mathrm{T}} + (2\tilde{q}-\tilde{p})\mathrm{Id}$. Inserting this into the matrix determinant, we find

$$G = \sqrt{\frac{N}{d\hat{\alpha}}}^n \left(q-p+\frac{N}{d\hat{\alpha}}\right)^{-\frac{n}{2}} \left(1+\frac{np}{q-p+\frac{N}{d\hat{\alpha}}}\right)^{-\frac{1}{2}}$$

and

$$S_i(\tilde{Q}) = -\frac{n-1}{2}\ln|1+J_ig(\lambda_i)+\lambda_i\,(2\tilde{q}-\tilde{p})| - \frac{1}{2}\ln|1+J_ig(\lambda_i)+\lambda_i\,(2\tilde{q}+(n-1)\tilde{p})| - n\ln\lambda_i$$

We now find and solve the saddle point equations: $\frac{d}{da}S = 0$ for $a \in \{q, p, \tilde{q}, \tilde{p}\}$ for $n \in (0,\infty)$. This yields the following four conditions:

$$\begin{aligned}
\tilde{q} &= \frac{N}{2dn}\left(\frac{(n-1)}{q-p+\frac{N}{d\hat{\alpha}}} + \frac{1}{q+(n-1)p+\frac{N}{d\hat{\alpha}}}\right)\\
\tilde{p} &= \frac{N}{dn}\left(-\frac{1}{q-p+\frac{N}{d\hat{\alpha}}} + \frac{1}{q+(n-1)p+\frac{N}{d\hat{\alpha}}}\right)\\
q &= \frac{1}{dn}\sum_i\left[\frac{(n-1)\lambda_i}{1+J_ig(\lambda_i)+\lambda_i\,(2\tilde{q}-\tilde{p})} + \frac{\lambda_i}{1+J_ig(\lambda_i)+\lambda_i\,(2\tilde{q}+(n-1)\tilde{p})}\right]\\
p &= \frac{1}{nd}\sum_i\left[\frac{-\lambda_i}{1+J_ig(\lambda_i)+\lambda_i\,(2\tilde{q}-\tilde{p})} + \frac{\lambda_i}{1+J_ig(\lambda_i)+\lambda_i\,(2\tilde{q}+(n-1)\tilde{p})}+\right]
\end{aligned}$$

To simplify these expressions, we make the following observations. First $\tilde{q}, \tilde{p}$ only depend on $u := q-p$ and $w := q+(n-1)p$. Second $q, p$ only depend on $\tilde{u} = 2\tilde{q}-\tilde{p}$ and $\tilde{w} = 2\tilde{q}+(n-1)\tilde{p}$. It is hence possible to re-parametrize the problem and thereby decouple some of the variables. Concretely, using the first two equations, we find

$$\tilde{u} = \frac{N}{d}\frac{1}{u+\frac{N}{d\hat{\alpha}}} \qquad\qquad \tilde{w} = \frac{N}{d}\left(\frac{1}{w+\frac{N}{d\hat{\alpha}}}\right)$$

The second two equations yield

$$u = \frac{1}{d}\sum_i\left[\frac{\lambda_i}{1+J_ig(\lambda_i)+\lambda_i\tilde{u}}\right] \qquad\qquad w = \frac{1}{d}\sum_i\left[\frac{\lambda_i}{1+J_ig(\lambda_i)+\lambda_i\tilde{w}}\right]$$

This defines a self-consistency equation each for $u, w$. Interestingly, both pairs of self-consistency equations are the same. Assuming that the solution is unique, we find that $u = w$,$\tilde{u} = \tilde{w}$ and hence $p, \tilde{p} = 0$. With this, we find that $q$ must be found self-consistently from

$$q(J) = \frac{1}{d}\sum_i\left[\frac{\lambda_i}{1+J_ig(\lambda_i)+\lambda_i\left(\frac{N}{d}\frac{1}{q+\frac{N}{d\hat{\alpha}}}\right)}\right] \tag{47}$$

and

$$\tilde{q} = \frac{N}{2dn}\frac{1}{q+\frac{N}{d\hat{\alpha}}}$$

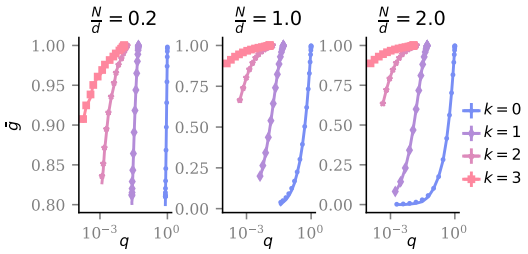



Figure 8: Relation between $\bar{g}$ and $q$. Full lines are predictions using eq. (55), markers show simulations for different fractions of $N/d$, and varying values of $\hat{\alpha}$ between $2^{-9}, 2^{9}$. Colors and marker styles distinguish and different spectra of the true covariance matrix $\Sigma$, namely $\lambda_i = i^{-k}\, \forall i = 1, \ldots, d$.

### E.3 Taking the limit $n \to 0$

Inserting the saddle-point values for $q, \tilde{q}, p, \tilde{p}$, we find

$$\begin{aligned}\ln \bar{Z}^n(J) =& dn \left[ \frac{N}{2d} \frac{q(J)}{q(J) + \frac{N}{d\hat{\alpha}}} - \frac{1}{2d} \sum_i \left[ \ln \left| 1 + J_i g(\lambda_i) + \lambda_i \left( \frac{N}{d} \frac{1}{q(J) + \frac{N}{d\hat{\alpha}}} \right) \right| \right] \right. \\ & \left. + \frac{N}{2d} \ln \frac{N}{d\hat{\alpha}} - \frac{N}{2d} \ln \left( q(J) + \frac{N}{d\hat{\alpha}} \right) \right]\end{aligned}$$

Due to the linear appearance of $n$ in $\ln \bar{Z}^n(J)$, we can finally take the limit $n \to 0$ in eq. (39) and find the expression for $f$

$$f(J) = \frac{N}{2d} \frac{q(J)}{q(J) + \frac{N}{d\hat{\alpha}}} - \left[ \frac{1}{2d} \sum_i \ln \left| 1 + J_i g(\lambda_i) + \lambda_i \left( \frac{N}{d} \frac{1}{q(J) + \frac{N}{d\hat{\alpha}}} \right) \right| \right] - \frac{N}{2d} \ln \left( \frac{d\hat{\alpha}}{N} q(J) + 1 \right) . \tag{48}$$

Before we move on to take the derivative of $f$ with respect to $J$, we now check that our result is consistent with one result from random matrix theory. To do so, we will first make a relation between $q$ and the following expression

$$-2 \sum_i \left. \frac{df}{dJ_i} \right|_{J=0, g(x)=x} = \frac{1}{d} \left\langle \mathrm{Tr} \Sigma \frac{1}{\mathrm{Id} + \hat{\alpha} \Sigma_0} \right\rangle_{\Sigma_0} = \frac{1}{d} \sum_i \frac{\lambda_i}{1 + \lambda_i \left( \frac{N}{d} \frac{1}{q + \frac{N}{d\hat{\alpha}}} \right)} = q \tag{49}$$

where the first equality follows from the definition of $f$, see eq. (38), the second follows from eq. (48) and the final equality is a consequence of eq. (47).

### E.4 Consistency checks

To validate our result, we perform a consistency check, using on prior knowledge on the Stieltjes transform $g$ of the random matrix $\Sigma_0 = \Sigma^{\frac{1}{2}} \mathcal{W} \Sigma^{\frac{1}{2}}$, defined by

$$g(z) = \lim_{d \to \infty} \frac{1}{d} \mathrm{Tr} \frac{1}{z \mathrm{Id} - \Sigma^{\frac{1}{2}} \mathcal{W} \Sigma^{\frac{1}{2}}}$$

where $\mathcal{W}$ is a Wishart with parameter $\frac{d}{N}$. It is known that in this case, $g$ fulfills the following self-consistency equation:

$$g(z) = \int d\rho_\Sigma(\lambda) \frac{1}{z - \lambda \left( 1 - \frac{d}{N} + \frac{d}{N} z g(z) \right)} \tag{50}$$

where $\rho_\Sigma(\lambda)$ is the spectral density of $\Sigma$, $\rho_\Sigma(\lambda) = \frac{1}{d} \sum_i \delta(\lambda - \lambda_i)$ in our case. Defining

$$\bar{g} = -\frac{1}{\hat{\alpha}} g \left( -\frac{1}{\hat{\alpha}} \right) = \frac{1}{d} \mathrm{Tr} \frac{1}{\mathrm{Id} + \hat{\alpha} \Sigma^{\frac{1}{2}} \mathcal{W} \Sigma^{\frac{1}{2}}} \tag{51}$$

this variable then fulfills the self-consistency equation

$$\bar{g} = \frac{1}{d} \sum_i \frac{1}{1 + \hat{\alpha}\lambda_i \left(1 - \frac{d}{N} + \frac{d}{N}\bar{g}\right)} \,. \tag{52}$$

We will now relate the self-consistency relation for $g$ to the self-consistency equation for $q$. Observe that, from the replica calculation, it follows that

$$\bar{g} = \sum_i \frac{d}{dJ_i} f(J)\bigg|_{J=0} = \frac{1}{d} \sum_i \frac{1}{1 + \lambda_i \left(\frac{1}{\hat{\alpha}\frac{d}{N}q + 1}\right)} =: s(q)$$

Using the replica self-consistency relation for $q$, for $J = 0$,eq. (47), we find

$$s(q) := \frac{\hat{\alpha}\left(\frac{d}{N} - 1\right) q + 1}{\hat{\alpha}\frac{d}{N}q + 1} \,, \tag{53}$$

Furthermore, it follows from the definition eq. (53), that

$$1 - \frac{d}{N} + \frac{d}{N} s(q) = \frac{1}{\hat{\alpha}\frac{d}{N}q + 1}$$

which, inserted into the definition of $s$, yields a self-consistency equation for $s(q)$,

$$s(q) = \frac{1}{d} \sum_i \frac{1}{1 + \hat{\alpha}\lambda_i \left(1 - \frac{d}{N} + \frac{d}{N} s(q)\right)} \tag{54}$$

which is identical to the self-consistency equation for $\bar{g}$, eq. (52), meaning that the replica calculation produces a variant of the known self-consistency relation eq. (50) for the Stieltjes transform. Furthermore the replica calculation yields a relation between $q = \frac{1}{d}\left\langle \mathrm{Tr}\left(\Sigma \frac{1}{1+\hat{\alpha}\Sigma_0}\right)\right\rangle_{\Sigma_0}$ and $\bar{g} = \frac{1}{d}\left\langle \mathrm{Tr}\left(\frac{1}{1+\hat{\alpha}\Sigma_0}\right)\right\rangle_{\Sigma_0}$ which is independent of the spectrum of $\Sigma$, namely

$$\bar{g} = s(q) \tag{55}$$

We test this relation in fig. 8, finding excellent agreement with simulations.

### E.5 Squared difference on test examples

We compute the generalization measure for fixed $t$:

$$\langle \left|\left| \left(W_t^* - W_t^{\text{oracle}}\right) x_t \right|\right|^2 \rangle_{x_t, \Sigma_0} = \frac{1 - \bar{\alpha}_t}{(1 - \bar{\alpha}_t + \gamma_t)} \left( \frac{\psi_{1,1} + \psi_{1,2}}{(1 - \bar{\alpha}_t + \gamma_t)} - 2\psi_2 + \psi_3 \right)$$

where we defined

$$\psi_{1,1}^t = \left\langle \mathrm{Tr}\left[\frac{1 - \bar{\alpha}_t}{\left(\mathrm{Id} + \hat{\alpha}_t\Sigma_0\right)^2}\right]\right\rangle \quad \psi_{1,2}^t = \left\langle \mathrm{Tr}\left[\frac{\bar{\alpha}_t}{\left(\mathrm{Id} + \hat{\alpha}_t\Sigma_0\right)^2}\Sigma\right]\right\rangle \quad \psi_2^t = \left\langle \mathrm{Tr}\left[\frac{1}{\left(\mathrm{Id} + \hat{\alpha}_t\Sigma_0\right)}\right]\right\rangle \tag{56}$$

We first compute $\psi_2$, using that

$$\psi_2 = -2 \sum_i \frac{d}{dJ_i} f(J)\bigg|_{J=0, g=1}$$

We now find $\frac{\partial f(J)}{\partial q} = 0$, hence $\frac{df}{dJ_i} = \frac{\partial f(J)}{\partial J_i}$, which is equal to

$$\frac{df}{dJ_i} = -\frac{1}{2d} \frac{g(\lambda_i)}{1 + J_i g(\lambda_i) + \lambda_i \left(\frac{N}{d}\frac{1}{q + \frac{N}{d\hat{\alpha}}}\right)}$$

Where $q$ is found, e.g. by solving eq. (52).

For $\psi_{1,1}$and $\psi_{1,2}$, we first note that for precision matrix $A$, and diagonal $J, \Lambda$, we have that

$$\frac{d^2}{dJ_i dJ_j} \ln \int \frac{d\eta}{\sqrt{2\pi}^d} \exp\left(-\frac{1}{2}\eta^{\mathrm{T}}\left[A + Jg(\Lambda)\right]\eta\right)\Big|_{J=0} = \frac{f(\Lambda)_{ii} f(\Lambda)_{jj}}{2}\left(A^{-1}_{ij}\right)^2$$

Where we used Wick's theorem to evaluate the Gaussian moments. For the specific functions of interest eq. (56), we find

$$\psi_{1,1} = 2\sum_{ij}\left(\frac{d^2}{dJ_i dJ_j} f(J)\right)\Big|_{J=0,g(x)=1} \qquad \psi_{1,2} = 2\sum_{ij}\left(\frac{d^2}{dJ_i dJ_j} f(J)\right)\Big|_{J=0,g(x)=\sqrt{x}}$$

the only difference between the two being the function $g$. Hence, we compute the second derivatives

$$\frac{d^2 f}{dJ_i dJ_j}\Big|_{J=0} = \delta_{ij}\,\frac{1}{2d}\frac{g^2(\lambda_i)}{\left(1+\lambda_i\left(\frac{N}{d}\frac{1}{q+\frac{N}{d\hat{\alpha}}}\right)\right)^2} - \frac{N}{2d^2}\frac{g(\lambda_i)\lambda_i}{\left(q+\frac{N}{d\hat{\alpha}}+\frac{N}{d}\lambda_i\right)^2}\frac{dq}{dJ_j}\Big|_{J=0}$$

Using the self-consistency equation, we find

$$\frac{dq}{dJ_i}\Big|_{J=0} = -\frac{\left(q+\frac{N}{d\hat{\alpha}}\right)^2}{d\left(1-\frac{N}{d}R_2(q)\right)}\frac{\lambda_i g(\lambda_i)}{\left(q+\frac{N}{d\hat{\alpha}}+\frac{N}{d}\lambda_i\right)^2}$$

where we defined

$$R_k(q) = \frac{1}{d}\sum_j \frac{\lambda_j^k}{\left(q+\frac{N}{d\hat{\alpha}}+\frac{N}{d}\lambda_j\right)^2}\,.$$

Putting it all together, for the specific functions of interest, we find

$$\psi_{1,1} = \left(q+\frac{N}{d\hat{\alpha}}\right)^2\left[R_0 + \frac{N}{d}\frac{R_1(q)^2}{\left(1-\frac{N}{d}R_2(q)\right)}\right] \quad \psi_{1,2} = \left(q+\frac{N}{d\hat{\alpha}}\right)^2\left[R_1 + \frac{N}{d}\frac{R_{\frac{3}{2}}(q)^2}{\left(1-\frac{N}{d}R_2(q)\right)}\right]$$

and

$$\psi_2 = \frac{1}{d}\sum_i \frac{1}{1+\lambda_i\left(\frac{N}{d}\frac{1}{q+\frac{N}{d\hat{\alpha}}}\right)}\,.$$

### E.6 Residual and test loss at finite $N$

To compute the residual loss, we employ the following identity:

$$\begin{aligned}\frac{df}{d\hat{\alpha}_t}\Big|_{J=0,g=1} &= -\frac{1}{2Nd}\sum_{\beta=1}^{N}\left\langle \sum_{ij}\left(\Sigma^{\frac{1}{2}}x^\beta\right)_i\left(\frac{1}{1+\hat{\alpha}\Sigma_0}\right)_{ij}\left(\Sigma^{\frac{1}{2}}x^\beta\right)_j\right\rangle \\ &= -\frac{1}{2d}\left\langle \mathrm{Tr}\Sigma_0\frac{1}{\mathrm{Id}+\hat{\alpha}\Sigma_0}\right\rangle\end{aligned}$$

Inserting this into the equation for the residual loss, eq. (22), which yields

$$\begin{aligned}R &= \frac{-2}{T}\sum_t \frac{\bar{\alpha}_t}{1-\bar{\alpha}_t+\gamma_t}\frac{df}{d\hat{\alpha}_t}\Big|_{J=0,g=1} + \frac{\gamma_t}{1-\bar{\alpha}_t+\gamma_t}\psi_2 \\ &= \frac{1}{T}\sum_t \frac{\bar{\alpha}_t}{1-\bar{\alpha}_t+\gamma_t}\frac{q}{\frac{d\hat{\alpha}_t}{N}q+1} + \frac{\gamma_t}{1-\bar{\alpha}_t+\gamma_t}\frac{1}{d}\sum_i \frac{1}{1+\lambda_i\left(\frac{N}{d}\frac{1}{q+\frac{N}{d\hat{\alpha}_t}}\right)}\end{aligned} \tag{57}$$

Second, we find that the test loss simplifies to

$$L_{test} = 1 + \frac{1}{T}\sum_t \frac{1-\bar{\alpha}_t}{(1-\bar{\alpha}_t+\gamma_t)}\left(\frac{\psi_{1,1}+\psi_{1,2}}{(1-\bar{\alpha}_t+\gamma_t)} - 2\psi_2\right)$$

which contains only functions which we have already computed in appendix E.5.

### E.7 Kullback-Leibler divergence

We compare $\rho = \mathcal{N}(\mu, \Sigma)$ to $\rho_N = \mathcal{N}(\mu_0, \Sigma_0 + \gamma \mathrm{Id})$ The DKL between two Gaussians is given by

$$\mathrm{DKL}(\rho_N|\rho) = \frac{1}{2}\left[\ln\frac{|\Sigma|}{|\Sigma_0 + c\mathrm{Id}|} + (\mu-\mu_0)^{\mathrm{T}}\Sigma^{-1}(\mu-\mu_0) + \mathrm{Tr}\Sigma^{-1}\left(\Sigma_0 + c\mathrm{Id}\right) - d\right], \quad (58)$$

We now average this expression over draws of the data set term by term. First, note that

$$\mathrm{Tr}\Sigma^{-1}\left(\langle\Sigma_0\rangle + c\mathrm{Id}\right) - d = c\,\mathrm{Tr}\Sigma^{-1} \quad (59)$$

Second, we compute

$$\begin{aligned}\left\langle(\mu-\mu_0)^{\mathrm{T}}\Sigma^{-1}(\mu-\mu_0)\right\rangle &= \frac{1}{N^2}\sum_{i,j,k,l=1}^{d}\sum_{\beta_1,\beta_2=1}^{N}\left\langle x_i^{\beta_1}x_j^{\beta_2}\right\rangle\left(\Sigma^{\frac{1}{2}} + \sqrt{c}\mathrm{Id}\right)_{ki}\Sigma^{-1}_{kl}\left(\Sigma^{\frac{1}{2}} + \sqrt{c}\mathrm{Id}\right)_{lj} \\ &= \frac{d + 2\sqrt{c}\mathrm{Tr}\Sigma^{-\frac{1}{2}} + c\mathrm{Tr}\Sigma^{-1}}{N}\end{aligned}$$

with $x^\beta \sim \mathcal{N}(0, \mathrm{Id})$. Finally, we have that when $c = \frac{1}{\hat{\alpha}}$

$$-\frac{1}{2}\left\langle\ln\left|\Sigma_0 + c\mathrm{Id}\right|\right\rangle = d\,f(J=0) + \frac{d}{2}\ln\hat{\alpha} \quad (60)$$

we now evaluate eq. (48) at $J = 0$ to find

$$-\frac{1}{2}\left\langle\ln\left|\Sigma_0 + \hat{\alpha}^{-1}\mathrm{Id}\right|\right\rangle = \frac{N}{2d}\frac{q}{q + \frac{N}{d\hat{\alpha}}} - \left[\frac{1}{2d}\sum_i \ln\left|\frac{1}{\hat{\alpha}} + \lambda_i\left(\frac{N}{d}\frac{1}{q\hat{\alpha} + \frac{N}{d}}\right)\right|\right] - \frac{N}{2d}\ln\left(\frac{d\hat{\alpha}}{N}q + 1\right)$$

such that all in all, the DKL simplifies to eq. (7).

### E.8 Bounds and approximations of $q$ and the KL divergence

#### E.8.1 Bounding $q$ from above

We now use two different approaches to bound $q$ from above. Defining

$$h(q) = \frac{1}{\frac{d}{N}q + \frac{1}{\hat{\alpha}}}$$

we find that

$$q = \frac{1}{d}\sum_{\nu=1}^{d}\frac{\lambda_\nu}{h(q)\lambda_\nu + 1} = h^{-1}(q)\frac{1}{d}\sum_{\nu=1}^{d}\frac{\lambda_\nu}{\lambda_v + h^{-1}(q)}$$

First, note that from eq. (49) follows that $q > 0$ hence, $h^{-1}(q) > 0$. With this, we can make a very coarse approximation that

$$q \geq \frac{1}{h(q)} \Rightarrow \left(1 - \frac{d}{N}\right)q \geq \frac{1}{\hat{\alpha}}$$

For $d < N$, this bounds $q$ from above via

$$d < N \quad \Rightarrow \quad q \leq \frac{1}{\hat{\alpha}\left(1 - \frac{d}{N}\right)} \quad (61)$$

For $\hat{\alpha}$ very large, one can show that the difference to the right hand side (see appendix E.8.3) is of order $\mathcal{O}\left(\hat{\alpha}^{-2}\right)$. This bound is only valid when $N > d$, additionally, it does not depend on the dimension. When $N < d$, we may instead use

$$q \leq \frac{1}{d}\mathrm{Tr}\Sigma \quad (62)$$

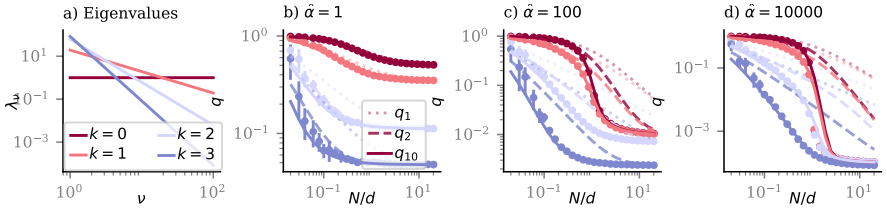



Figure 9: Comparison of $q$ to the upper bound. a) shows the spectra of the covariances we consider, $\lambda_\nu = c_k \nu^{-k}$ where we choose $c_k$ such that $\frac{1}{d}\sum_\nu \lambda_\nu = 1$. b)-f) compare numerical values of $q$ found using eq. (49) with $d = 100$. Dots and error bars denote mean and standard deviations over ten realizations of $\Sigma_0$, respectively. Dotted, dashed and full lines show upper bounds eq. (63) for increasing $n$.

this approximation itself is quite coarse. However, we may reinsert it into the equation for $q$ to obtain a smaller upper bound

$$h(q) \geq \frac{1}{\frac{d}{N}\frac{1}{d}\mathrm{Tr}\Sigma + \frac{1}{\hat{\alpha}}} \Rightarrow q \leq \left(\frac{d}{N}\frac{1}{d}\mathrm{Tr}\Sigma + \frac{1}{\hat{\alpha}}\right) \frac{1}{d}\mathrm{Tr}\frac{\Sigma}{\Sigma + \mathrm{Id}\left(\frac{d}{N}\frac{1}{d}\mathrm{Tr}\Sigma + \frac{1}{\hat{\alpha}}\right)}$$

which is a slightly tighter bound on $q$, which we can reinsert into the expression for $q$ again to obtain an even smaller upper bound. We may alternatively define the following series

$$q_0 = \frac{1}{d}\mathrm{Tr}\Sigma, \quad q_{n+1} = \left(\frac{d}{N}q_n + \frac{1}{\hat{\alpha}}\right) \frac{1}{d}\mathrm{Tr}\frac{\Sigma}{\Sigma + \mathrm{Id}\left(\frac{d}{N}q_n + \frac{1}{\hat{\alpha}}\right)} \tag{63}$$

we then find that $q \leq q_n \forall n$. We compare numerical simulations of eq. (49) to the bound on $q$ obtained thus in fig. 9, finding that the bound becomes increasingly tight as we increase $n$.

### E.8.2 Faster spectral decay leads to smaller $q$

Following the intuition expressed in the main text, we here make an argument to show that a faster spectral decay implies a smaller $q$. Let us compare two covariance matrices with equal trace $\mathrm{Tr}\Sigma_1 = \mathrm{Tr}\Sigma_2$, but $\Sigma_1$ has a faster spectral decay than $\Sigma_2$. The objective is to show that this implies $q_1 < q_2$.

To show this, we consider the function $\bar{g}$ defined in eq. (51), which is related to $q$ via $s(q) = \bar{g}$ (eq. (55)). Since $s(q)$ is strictly decreasing as a function of $q$, it therefore suffices to show that $g_1 > g_2$ to imply that $q_1 < q_2$.

To see that $g_1 > g_2$, consider the vectors of eigenvalues $\lambda_1, \lambda_2$ corresponding to $\Sigma_1, \Sigma_2$. Then if $\Sigma_1$ has a faster spectral decay, we know that $\lambda_1$ majorizes $\lambda_2$. Moreover the self-consistency equation for $\bar{g}$, given by eq. (52), that relates the spectrum to $\bar{g}$, has a unique solution, as both sides of the equation are monotonous in $\bar{g}$. Then Karamata's inequality gives that the right hand side of eq. (52) is larger for $\Sigma_1$ than for $\Sigma_2$, therefore $g_1 > g_2$.

### E.8.3 $q$ at $N \gg d$ and large $\hat{\alpha}$

We now seek an approximation for $q$ in the regime $N \gg d$ and large $\hat{\alpha}$. To do so, we first examine $\bar{g} = \frac{1}{d}\left\langle \mathrm{Tr}\left(\frac{1}{1+\hat{\alpha}\Sigma_0}\right)\right\rangle_{\Sigma_0}$, which we found to relate to $q$ via the relation eq. (55). We now additionally assume that $\hat{\alpha}$ is much larger than $\left(\lambda^0_{\mathrm{min}}\right)^{-1}$, where $\lambda^0_{\mathrm{min}}$ is the smallest eigenvalue in $\Sigma_0$. This assumption is only valid for at least $N \geq d$ as otherwise $\Sigma_0$ has zero eigenvalues. Then it follows that $\bar{g}$ is of order $\hat{\alpha}^{-1}$. We now invert the relation between $q$ and $\bar{g}$, finding that

$$q = \frac{1}{\hat{\alpha}\frac{d}{N}}\left(\frac{1}{1 - \frac{d}{N} + \frac{d}{N}\bar{g}} - 1\right) \approx \frac{1}{\hat{\alpha}\frac{d}{N}}\left(\frac{1}{1 - \frac{d}{N}} - 1\right) \tag{64}$$

which is independent of $\Sigma$. Inserting this into eq. (7), we find

$$\frac{\mathrm{DKL}(\rho_N|\rho)}{d} = \frac{1 + \frac{N}{d}\ln(1 - \frac{d}{N}) + \ln(1 - \frac{d}{N}) + \frac{N}{d^2}}{2} + \mathcal{O}\left(\sqrt{\hat{\alpha}^{-1}}\right), \tag{65}$$

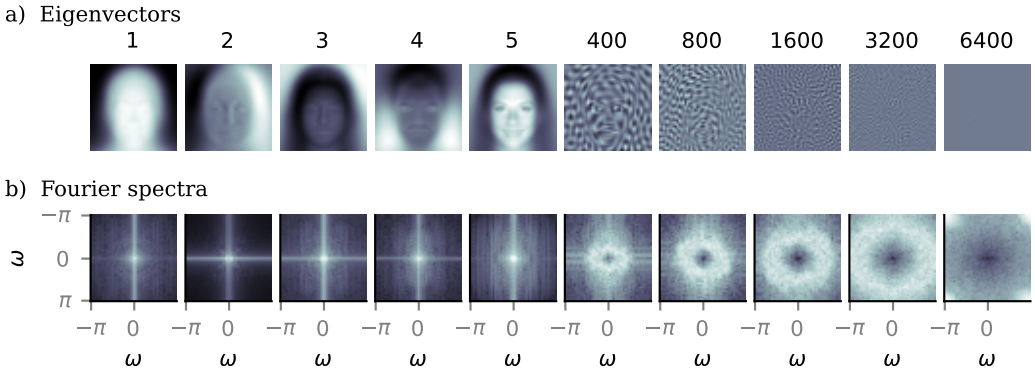



Figure 10: a) First five leading eigenvectors $\nu = 1, \ldots, 5$ of the covariance matrix of the CelebA data, as well as sub-leading eigenvectors $\nu = 400, \ldots, 6400$. b) Corresponding Fourier spectra of the eigenvectors.

which, for $N \gg d$, scales as $\frac{d}{4N}$.

## F A detail-based similarity measure for CelebA and CIFAR-10

For the models trained on image data, computing the cosine similarity $c(x, y) = \frac{x^{\mathrm{T}} y}{|x||y|}$ between a generated sample and one from the training set yields a very high similarity $\sim 0.9$, even if the generated images are genuinely different. Upon manual inspection of the corresponding images, we find that this occurs due to a large portion of the image, such as the background, being uniformly dark or light.

In fig. 10 we compare the eigenvectors of the covariance matrix of the CelebA data to their corresponding Fourier spectra. We find that leading eigenvectors $\nu = 1, \ldots, 5$ have a more homogeneous spatial distribution of light and dark pixels, correspondingly their Fourier spectra are concentrated around small frequencies (small $|\omega|$). As $\nu$ increases, however, the spectra of the eigenvectors become more broad, and small frequencies are suppressed.

On the basis of these observations, we construct a similarity measure which is oriented more towards the details of the images: We first project the images into the space spanned only by sub-leading eigenvectors $\nu > 5$. We then compute the cosine similarities of the resulting vectors. We find that this measure is then more sensitive to changes in the details of the images, which leave the background uniform (e.g. for the generated image and closest training set examples in fig. 1 for $N \geq 6400$).

## G Differences between linear and non-linear models

In fig. 11 we report the difference in the mapping of the CelebA compared to the best linear model. We observe qualitatively the same behavior as in the CIFAR-10 data: overall, the relative error decays with $N$, $\nu$ and $t$. Indeed for a large extent of $t, \nu$, the difference between linear and non-linear models becomes very small. However, for leading eigenmodes (small $\nu$), the differences between linear and non-linear models grow with $N$.

## H Nullspace effect in larger U-Nets

We here report the outcome of the experiment computing the memorization of trained non-linear diffusion models for the larger U-Net model class. For this architecture the memorization effect is dominant. We find that among $4,000$ generated samples, $1435$ are classified as non-memorized for the $N = 1000$ model, while only $103$ are classified as non-memorized for the $N = 500$ model. The number of non-memorized generated samples is small, making the corresponding nullspace estimates

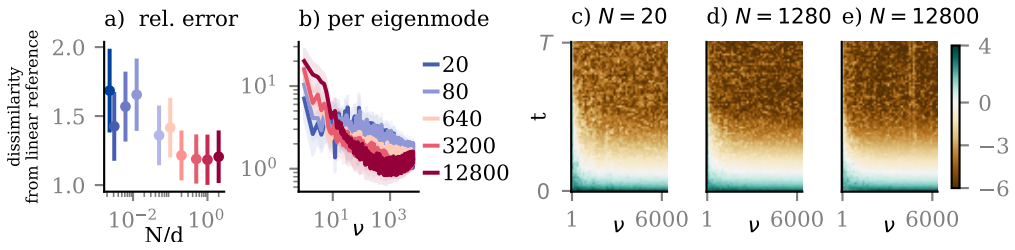



Figure 11: Relative difference of non-linear denoisers from best linear model, per noising step $t$ and direction $\nu$, trained on inecreasing numbers of data. a) averaged over $\nu$ and $t$, b) averaged over $t$, c) - e) $\log d_{t,\nu}$ per $\nu, t$. All data are averaged over 100 test samples per $t, \nu$.

less reliable. Nevertheless, the experiment with the reduced model size shows that the covariance overfitting effect can become relevant for models which do not have the capacity to memorize.

# I Nomenclature

Table 1: Nomenclature part 1.

| Symbol | Meaning | Notes / Context |
|---|---|---|
| $\rho$ | Population data distribution | Unknown target distribution |
| $\rho_t$ | Distribution of noised variables $x_t$ | Used in score matching |
| $\rho_N$ | Sampling distribution of finite-$N$ linear model | Equal to $\mathcal{N}(\mu_0, \Sigma_0 + cI_d)$ |
| $d$ | Data dimension | Ambient dimension of samples |
| $N$ | Number of training samples | Dataset size |
| $D$ | Training dataset | $\lvert D\rvert = N$ |
| $x_0$ | Clean data sample | Drawn from $\rho$ |
| $x_t$ | Noised sample at timestep $t$ | $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon_t$ |
| $\epsilon_t$ | Gaussian noise at timestep $t$ | $\epsilon_t \sim \mathcal{N}(0, I_d)$ |
| $t$ | Noising timestep | $t \in \{1, \dots, T\}$ or continuous time |
| $T$ | Number of noising steps | Also appears in averaged losses |
| $\alpha_t$ | Per-step noise-schedule coefficient | Used in sampling equations |
| $\bar{\alpha}_t$ | Cumulative signal coefficient | Decreases with $t$ |
| $\beta_t$ | Noise-schedule increment | Continuous version $\hat{\beta}(t)$ |
| $\hat{\beta}(t)$ | Rescaled continuous-time noise rate | Defined by $\hat{\beta}(t) = \beta_t/h$ |
| $h$ | Continuous-time step size | $h = T^{-1}$ |
| $\zeta(t)$ | Integrated noise rate | $\bar{\alpha}(t) = e^{-\zeta(t)}$ |
| $I_d$ | $d \times d$ identity matrix | Isotropic covariance/noise |
| $\mu$ | Population mean | Mean of $\rho$ |
| $\Sigma$ | Population covariance matrix | Covariance of $\rho$ |
| $\mu_0$ | Empirical mean | Computed from $D$ |
| $\Sigma_0$ | Empirical covariance matrix | Computed from $D$ |
| $\lambda_\nu$ | Eigenvalue of $\Sigma$ | Sorted by rank $\nu$ |
| $\lambda_\nu^0$ | Eigenvalue of $\Sigma_0$ | Empirical covariance eigenvalue |
| $e_\nu^0$ | Eigenvector of $\Sigma_0$ | Empirical covariance eigenvector |
| $\nu$ | Eigenvalue/eigenmode rank index | Used for covariance spectra |
| $k$ | Power-law spectral exponent | $\lambda_\nu \sim \nu^{-k}$ |
| $\bar{\lambda}$ | Mean eigenvalue of $\Sigma$ | $\bar{\lambda} = d^{-1}\operatorname{Tr}\Sigma$ |
| $c$ | Effective covariance cutoff / regularization scale | Appears in $\Sigma_0 + cI_d$ |
| $\gamma_t$ | $L^2$ regularization strength at timestep $t$ | Often chosen as $\gamma_t = \bar{\alpha}_t c$ or related variants |
| $\epsilon_\theta$ | Noise-prediction denoiser network | Trained to predict $\epsilon_t$ |
| $\theta$ | Trainable model parameters | Parameters of denoiser |
| $L$ | Noise-prediction training loss | Denoising score matching objective |
| $L_{\text{test}}$ | Test loss for noise prediction | Evaluated on unseen data |
| $R$ | Residual / irreducible training loss | Minimum training loss for linear denoiser |
| $f_\theta$ | Data-prediction denoiser | Trained to predict $x_0$ |
| $L_D$ | Data-prediction training loss | Alternative objective |
| $L_{\text{test},D}$ | Test loss for data prediction | Test counterpart of $L_D$ |
| $R_D$ | Residual loss for data prediction | Minimum value for $L_D$ |
| $W_t$ | Linear denoiser weight matrix | In $\epsilon_\theta(x_t, t) = W_t(x_t - \sqrt{\bar{\alpha}_t}b_t)$ |
| $b_t$ | Linear denoiser bias | Optimal value $b_t^* = \mu_0$ |
| $W_t^*$ | Optimal linear denoiser weights | Closed-form minimizer of $L$ |
| $V_t$ | Linear data-prediction weight matrix | Used for $f_\theta(x_t, t) = V_t x_t + r_t$ |
| $r_t$ | Linear data-prediction bias | Bias in $f_\theta$ |
| $V_t^*, r_t^*$ | Optimal data-prediction parameters | Closed-form minimizers of $L_D$ |
| $D_{\text{KL}}(\rho_N \lvert \rho)$ | Kullback–Leibler divergence | Distance between generated and target Gaussians |
| $\Delta\epsilon_N$ | Difference from reference denoiser | Measures closeness to population/reference model |
| $\epsilon_N$ | Denoiser trained on $N$ samples | Finite-data model |
| $\epsilon_\infty$ | Reference denoiser | Population-loss or largest-data model |
| $d_{t,\nu}$ | Relative difference from linear reference | Resolved by timestep and eigendirection |
| $\eta$ | Learning rate | Used in gradient descent dynamics |
| $\tau$ | Training time / optimization time | Continuous training-time variable |
| $\tau^*$ | Optimal stopping time | Time minimizing test loss |
| $\eta_{\mu,t}$ | Mode-dependent learning rate / relaxation rate | Controls exponential convergence of weights |
| $\mu, \nu$ | Mode indices | Used in weight/eigenbasis expansions |

Table 2: Nomenclature part 2.

| Symbol | Meaning | Notes / Context |
|---|---|---|
| $u_s$ | Sampling-process state | Reverse-time variable during generation |
| $s$ | Reverse sampling time index | Often $s = T - t$ or continuous $s \in [0, 1]$ |
| $\xi$ | Sampling noise | $\xi \sim \mathcal{N}(0, I_d)$ |
| $\sigma_t$ | Sampling noise scale | Appears in reverse process |
| $\hat{\sigma}(t)$ | Rescaled continuous sampling noise scale | $\hat{\sigma}(t) = \sigma_t/\sqrt{h}$ |
| $\mu_\theta(x_t, t)$ | Reverse-process mean map | Used in sampling update |
| $m(s)$ | Linear drift coefficient in sampling SDE | Direction-wise in eigenbasis of $\Sigma_0$ |
| $c(s)$ | Affine drift term in sampling SDE | Depends on $\mu_0$ |
| $Z_s$ | Wiener process | Continuous-time sampling noise |
| $\rho(u, s)$ | Density of sampling variable | Fokker–Planck description |
| $\mu_u(s)$ | Mean of sampling variable | Direction-wise sampling mean |
| $\sigma_u^2(s)$ | Variance of sampling variable | Direction-wise sampling variance |
| $q$ | Replica order parameter / self-consistent scalar | Controls averaged finite-$N$ statistics |
| $q(J)$ | Source-dependent replica order parameter | Used in generating functional |
| $J$ | Source matrix / source variable | Introduced for derivatives of generating functional |
| $g(\Lambda)$ | Spectral test function in replica calculation | Not to be confused with Stieltjes transform notation |
| $f(J)$ | Replica generating functional | Produces inverse-matrix moments |
| $R$ | Fixed rotation matrix diagonalizing $\Sigma$ | In $\Sigma = R\Lambda R^\top$; distinct from residual loss by context |
| $\Lambda$ | Diagonal eigenvalue matrix of $\Sigma$ | $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_d)$ |
| $x^\beta$ | Standard Gaussian latent sample in replica derivation | $\beta = 1, \ldots, N$ |
| $\beta$ | Sample index in replica derivation | Distinct from noise schedule $\beta_t$ |
| $\alpha$ | Replica index | Used in replica trick |
| $n$ | Number of replicas | Limit $n \to 0$ |
| $Q_{\alpha_1,\alpha_2}$ | Replica overlap matrix | Auxiliary variable |
| $\tilde{Q}_{\alpha_1,\alpha_2}$ | Conjugate overlap variable | Enforces definition of $Q$ |
| $q, p, \tilde{q}, \tilde{p}$ | Replica-symmetric overlap parameters | Diagonal/off-diagonal Ansatz parameters |
| $\bar{g}$ | Rescaled Stieltjes-transform quantity | Related to $q$ |
| $R_k(q)$ | Spectral sum in replica formulas | Appears in $\psi$ terms |
| $\psi_{1,1}^t, \psi_{1,2}^t, \psi_2^t, \psi_3^t$ | Averaged trace quantities | Used for test-loss and reference-distance formulas |

## 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: All claims made in the abstract and in the introduction are supported either by numerical experiments, analytical calculations, or both.

   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 openly discusses the limitations of its approach in the discussion (section 4) and provides empirical observations testing its limitations in section 3.

   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 paper provides the assumptions for all theoretical results. While we don't state formal theorems, we provide the complete and correct derivation of all results, which use standard methods of statistical physics.

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 analytically generated curves are fully specified by the formulas in the main text and appendices. For the U-net experiments, the datasets, sample sizes, metrics, and averaging procedures are described in the relevant figure captions and experimental sections.

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: Numerical experiments are described in sufficient detail for reproduction and are not dependent on details of the implementation. We plan to upload the code in the future.

   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: All experiments are discussed in sufficient detail.

   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: All numerical results include error bars as well as suitable information to how they were 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 linear-model experiments require modest compute. The U-net experiments require GPU training; the datasets, sample sizes, and averaging procedures are reported in the paper.

   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 conducted in the paper conforms, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [N/A]

    Justification: The papers contents are of foundational nature. We do not foresee any direct negative or positive societal impact.

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 produce models which have high risks of misuse.

    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 only assets used in the paper are the standard datasets and algorithms, which are properly cited in the references.

    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 release new assets.

    Guidelines:

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

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

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

    Answer: [N/A]

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

    Guidelines:

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

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

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

    Answer: [N/A]

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

    Guidelines:

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

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

16. **Declaration of LLM usage**

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

    Answer: [N/A]

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

    Guidelines:

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