Generalization Dynamics of Linear Diffusion Models

Claudia Merger
SISSA, Trieste
cmerger@sissa.it

Sebastian Goldt
SISSA, Trieste
sgoldt@sissa.it

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 hierarchical structure of real data. To understand whether such a hierarchical 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 hierarchical organization of variance in the data and regularization impacts 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 strongly hierarchical data structure, as well as regularization and early stopping help to prevent overfitting. For $N > d$, 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 [1, 2, 3] have become the state-of-the-art paradigm in generative AI, where they are trained to sample from an unknown distribution ρ based on a finite set of training data drawn from ρ. While the behavior of diffusion models that have learned this mapping accurately [4, 5, 6, 7, 8, 9] as well as the learning dynamics of diffusion models trained on unstructured data [10, 11, 12] has been studied recently , much less is known about their behavior when trained on finite, *structured* datasets. Ideally, diffusion models abstract the characteristic statistics of ρ from the training data using a neural network. Conventional learning theory dictates that one needs a number of samples N that is exponential in d to accurately approximate a arbitrarily complex function on a d-dimensional space, far more than available in practice. The success of diffusion models then appears to indicate that ρ is simple enough such that its key properties can be inferred with fewer data.

Many machine learning datasets exhibit hierarchical structure, where some features are more important than others. Theoretical work on diffusion models captures this in different ways. One approach assumes a random hierarchy model [13] for ρ, showing that generalization occurs when N scales polynomially with d, [14, 15], though it lacks quantitative predictions for specific datasets. Another assumes data lie on a low-dimensional manifold [16, 17, 18], implying sample complexity depends on the manifold dimension rather than the embedding dimension.This manifold hypothesis is ubiquitous in the theory of learning with neural networks [19]. However, estimating manifold dimension is difficult. Moreover, recent work shows a more refined picture of dimensionality in image data [20] that challenges the notion of these data lying on a lower dimensional manifold of homogeneous dimension in space.

In contrast to these approaches, we make use of one salient feature of ρ that can be determined directly from the data: its covariance spectrum. Covariance spectra inform us about the relative

EurIPS 2025 Workshop on Principles of Generative Modeling (PriGM)

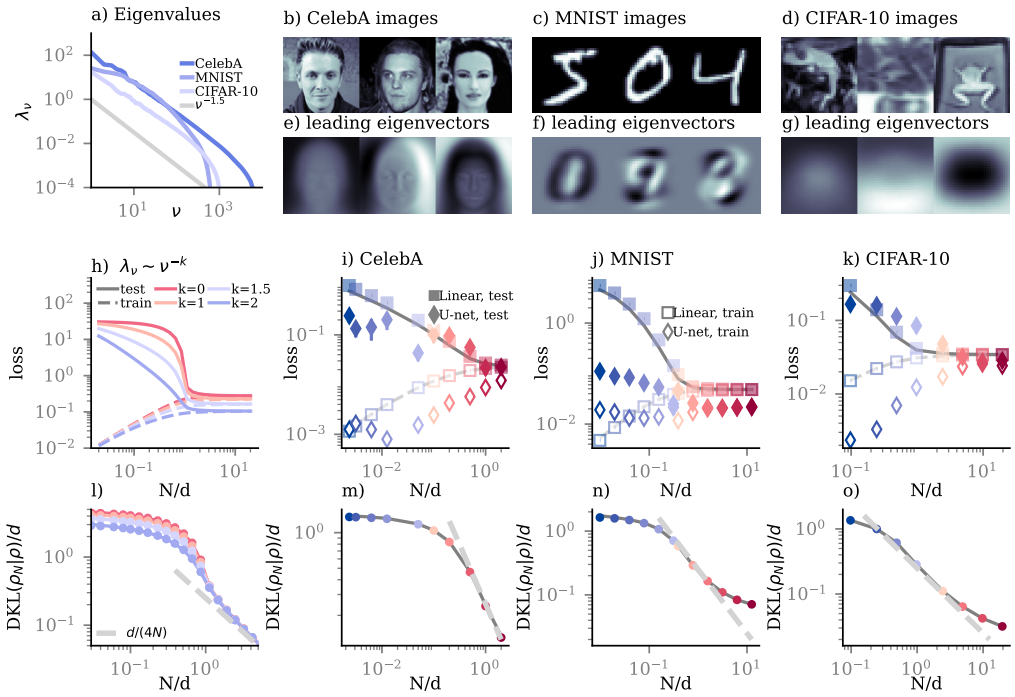

Figure 1: a) Eigenvalues of covariances obtained from image data sets, sorted by rank. b) - d) Example images from image datasets. e) - g) Top three leading eigenvectors of covariance matrices obtained from the full image dataset. h) Prediction for test and training loss of linear diffusion models trained on N samples $d = 100$-dimensional samples from $\mathcal{N}(0, \Sigma)$, where the eigenvalues of Σ 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 Σ 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 Σ originating from the CelebA, MNIST, and CIFAR-10 datasets, respectively. Grey lines show prediction eq. (5) from replica theory.

spread of the data distribution in different directions. For example, the covariance spectra of image data typically exhibit a power-law behavior, see fig. 1 a), where we show the covariance spectra for three popular image datasets. This hierarchy 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 informative features of the data, for example controlling for background color or the placement of shadows in an image, see fig. 1 d)-f). Furthermore, covariance spectra are known to affect learning dynamics in diffusion models: leading eigenmodes are typically learned faster than sub-leading ones [21, 22].

In this work, we investigate how hierarchical covariance spectra affect learning in diffusion models *in the undersampled regime*. We investigate how the undersampling of the covariance matrix due to limited data affects generalization in diffusion models. To this end, we will consider a linear neural networks that is able to produce samples with arbitrary covariance. Linear neural networks have helped elucidate overfitting in supervised learning in the past [23, 24] by providing a fully tractable case in which key mechanisms can be understood. A first analysis of linear diffusion models was given in [25]. Here, we extend their approach to include a hierarchical covariance structure that we fit to data. In the next section we give a brief introduction to diffusion models and we the neural network architecture we will study.

1 Linear Diffusion Models

Diffusion models consist of an iterative noising and denoising process. The noising process simplifies the distribution of a sample x_0 from ρ through the addition of noise

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

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 assumed to be close to $\mathcal{N}(0, \mathrm{Id})$. Diffusion models are neural networks ϵ_θ whose parameters θ are then optimized to approximate the score $\epsilon_\theta(x_t, t) \propto \nabla \ln \rho_t(x_t)$, where ρ_t is the density of the noised variables x_t

Typical architectures for ϵ_θ are very complex, including U-nets [26] and transformers. Here, we consider the case where for each t, the denoiser $\epsilon_\theta(\cdot, t)$ is an affine linear mapping, specified in appendix A whose weights we additionally regularize using a standard L^2 penalty whose strength γ_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 appendicesA and A.2. The learning outcomes of the linear model can be expressed using only the the empirical mean μ_0 and covariance Σ_0 of the training set $\mathcal{D}$ and the population mean μ and covariance Σ of ρ, 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.

2 Results

The nullspace of Σ_0 drives overfitting. As a first measure for generalization, we now compare the minimal training loss R to the test loss L_{test}, which quantifies the ability of the denoiser to denoise an unseen test example from ρ. We first define the eigendecomposition of the empirical covariance matrix Σ_0, with eigenvalues λ_ν^0, and normalized eigenvectors $\{e_\nu^0\}_\nu$. We find the gap between the R and L_{test} to be

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

This shows shows explicitly that the gap between R and L_{test} arises whenever there is a mismatch between μ_0, Σ_0 and μ, Σ. 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 λ_ν 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 $N - d$ eigenvalues of Σ_0 are exactly zero, giving rise to large contributions in eq. (2). This is reflected in a very large gap between training and test loss in fig. 1 h)-j) when $N < d$. However, this gap can effectively be reduced by regularization, through the presence of γ_t in the denominator of 2. We compare our results using linear models to U-nets trained on image data in fig. 1 h) - j). We find a similar saturation at $N \sim d$, but U-nets are naturally able to outperform linear models at large N.

We now move to a measure which directly compares the distributions of the generated samples ρ_N, and ρ, the Kullback-Leibler divergence (DKL). In appendix B, we show that $\rho_N = \mathcal{N}(\mu_0, \Sigma_0 + c\mathrm{Id})$, 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 (see appendix B for details), or from the regularizing with a particular choice $\gamma_t = \sqrt{\bar{\alpha}_t} c$ for the regularization strength.

We now impose a Gaussian hypothesis on the data $\rho = \mathcal{N}(\mu, \Sigma)$. This assumption lets us treat exclusively the deviations between ρ, ρ_N which arise due to finite N.To characterize the deviations due to finite N, we compute the DKL between the distribution of samples and ρ

$$\mathrm{DKL}(\rho_N|\rho) = \frac{1}{2}\left[\ln \frac{|\Sigma|}{|\Sigma_0 + c\mathrm{Id}|} + (\mu - \mu_0)^{\mathrm{T}} \Sigma^{-1} (\mu - \mu_0) + \mathrm{Tr}\Sigma^{-1}\left(\Sigma_0 + c\mathrm{Id}\right) - d \right] , \tag{3}$$

which is a measure of distance between distributions. The most dominant term at small N is $\mathrm{DKL}(\rho_N|\rho)$ is $\ln \frac{|\Sigma|}{|\Sigma_0^{\text{eff.}}|} = \sum_\nu \ln \frac{\lambda_\nu}{\lambda_\nu^0 + c}$, where λ_ν are the eigenvalues of Σ. This term in the DKL heavily penalizes the presence of a nullspace in Σ_0 for small c, analogous to the test loss.

Hierarchical spectra mitigate overfitting. Eqs. 3 show that the penalty for a nullspace of Σ_0 is less severe when Σ has small eigenvalues, hence when its spectrum is more hierarchical. To capture the effect of a hierarchical spectrum of Σ, we compute the average of the test loss and DKL in the average over draws of the training set from which Σ_0 is computed. We use the replica trick to evaluate the necessary averages; the full calculation is in found in appendix D. For brevity, we will focus on the typical case analysis of the DKL here. Our results depend on a a quantity q which must be determined self-consistently from the eigenvalues $\{\lambda_\nu\}_{\nu=1}^d$ of Σ and c

$$q = \frac{1}{d}\sum_\nu \frac{\lambda_\nu}{1+\lambda_\nu \frac{N}{dq+Ndc}} \tag{4}$$

In the average over draws of the training set, the DKL is given by

$$\begin{aligned}\frac{1}{d}\mathrm{DKL}(\rho_N|\rho) =& \frac{1}{2}\frac{q}{\frac{d}{N}q+c} - \frac{1}{2d}\sum_i \ln\left|\frac{c}{\lambda_i}+\frac{1}{\frac{d}{Nc}q+1}\right| - \frac{N}{2d}\ln\left(\frac{d}{Nc}q+1\right) \\ &+ \frac{d+2\sqrt{c}\mathrm{Tr}\Sigma^{-\frac{1}{2}}+c(N+1)\mathrm{Tr}\Sigma^{-1}}{2Nd}\end{aligned} \tag{5}$$

In fig. 1 k) -l) we compare the prediction obtained from eq. (5) to numerical simulations, showing excellent agreement. The first line in eq. (5) originates from the term $\ln|\Sigma|/|\Sigma_0+c\mathrm{Id}|$ and dominates 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. In appendix D.8 we also show that

$$q \leq \left(\frac{d}{N}\bar{\lambda}+c\right)\frac{1}{d}\mathrm{Tr}\frac{\Sigma}{\Sigma+\mathrm{Id}\frac{d}{N}\left(\bar{\lambda}+c\right)} \tag{6}$$

where $\bar{\lambda}=\frac{1}{d}\mathrm{Tr}\Sigma$ is the average over the eigenvalues of Σ. At fixed d/N, the more hierarchical the spectrum, i.e. the more eigenvalues of Σ are significantly smaller than average, the smaller q will be, and therefore, the smaller the DKL. We show examples of this both for spectra which are explicitly powerlaw, $\lambda_\nu \sim \nu^{-k}$ and Σ 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). Since eq. (5) decreases with q, this demonstrates how a more hierarchical spectrum can lead to a better fit. Intuitively, this is because the absence of variation in Σ_0 is not as significant when the corresponding variation in Σ is also small. On the other hand, for $N > d$, the DKL collapses on to the same line independently of the specifics of Σ. The independence of the DKL on Σ has been noted in [27], who argued that this makes the DKL a good measure for the similarity of Σ_0 to Σ. In D.8.2, we show that when c is much smaller than the smallest eigenvalue of Σ 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 Σ (see fig. 1 k) - l)), up to deviations which originate from $c > 0$, which causes a saturation of the DKL above zero.

Convergence to a reference model. We now turn to a different comparison between diffusion models which 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^{test}) are noised test samples, ϵ_N^t is the mapping obtained from a finite dataset of N samples and ϵ_∞ is the mapping obtained from an infinite number of samples. This measure compares the mappings implemented by a diffusion model optimized on finite N to the best reference. In contrast to the test loss, this measure does not saturated above $N \sim d$ and can therefore also measure how non-linear models approach their optima in this regime. In practice, when only finite number of data are available, we choose ϵ_∞ as the mapping optimized on the largest subset of the data. In fig. 2 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 D.5). We find that both for linear and non-linear diffusion models, this difference from the reference model decreases significantly at linear sample complexity.

Memorization When N is small, we find that the nonlinear models memorize the data, i.e. that new samples generated from the trained models are almost identical to training samples. In 2. To measure memorization, we find the closest training image for each generated image based on a detail-based similarity measure (defined in appendix E). The similarity between generated and training set examples is highest at small N and decreases as N increases, see e.g. [28, 12, 14]. We find that in our experiments, memorization diminishes when $N \sim d$. As early stopping has been shown to prevent memorization [14, 12], the point where we observe memorization to diminish could shift to larger N when the models are trained longer.

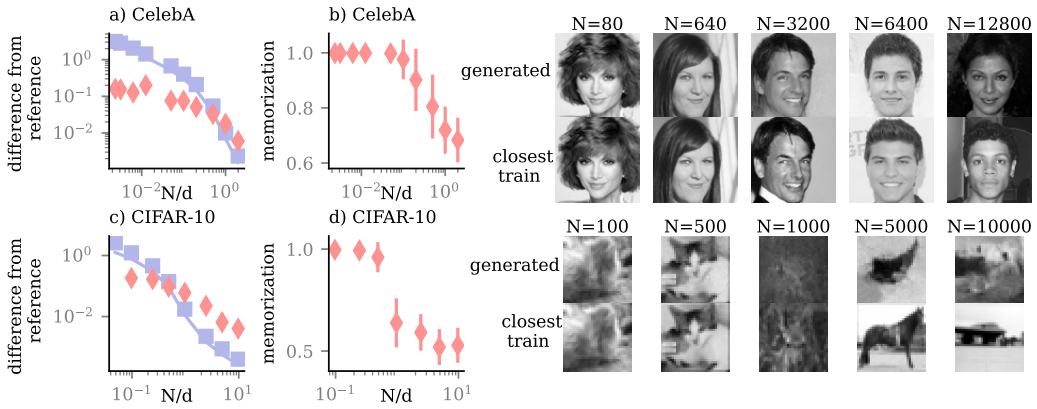

Figure 2: Left column: a) Difference of denoisers from reference model, trained on increasing numbers of data, averaged over 100 test data points. Blue squares are linear models, pink diamonds are U-nets. Blue lines show prediction from replica calculation. b) Similarity between samples generated from 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 inctreasing number of data.

Differences between linear and non-linear models. Recent experimental studies [29, 30]suggest that diffusion models closely approximate linear models when N is large. In appendix F, we test this hypothesis for different levels of noise and in different spatial directions ν given by the eigenvectors of Σ. We find that for a large extent of t, ν, the difference between linear and non-linear models diminishes with N. However, for leading eigenmodes (small ν), the difference between linear and non-linear models grows with N. This (small ν, small t) is also the regime where we expect the non-Gaussianity of the data to have the largest effect.

3 Discussion

We have identified two relevant regimes for generalization in linear diffusion models, $N > d$ and $N < d$. At $N > d$, we observe a saturation in the test loss, and a decay of the Kullback-Leibler divergence that is independent of the data structure. An in-depth treatment of this regime for energy-based models is given by Catania et al. [21], who show that early stopping and regularization can help mitigate overfitting; the same holds also for linear diffusion models. When $N < d$, the model overfits due to a lack of variability in the training set, namely the low-rank structure of the empirical covariance matrix. Both the Kullback-Leibler divergence and test loss strongly penalize this lack of variability. In this regime, a hierarchical data structure that is typical of image data is beneficial for learning. The more hierarchical a dataset is, the lower the test error and Kullback-Leibler divergence will be. The presence of regularization in the form of c can be interpreted as placing a cutoff on the minimal variation of the data in any direction, below which the structure of the covariance will no longer be resolved. Hence c plays the role of the relevant scale of the data, a concept that echoes previous investigations relating principal component analysis and deep learning to the renormalization group [31, 32, 33, 34, 35].

Intriguingly, we found that a highly hierarchical structure in the data has no significant effect on the emergence of the two regimes, i.e. the number of data N where these two regimes intersect. This suggests that diffusion models place emphasis on learning all directions with finite variability, not only those with the highest levels of variation. A similar effect has previously been observed in [36], where learning in the supervised setting was contrasted with diffusion models. In the supervised case, an effect called benign overfitting occurs: if the data consist of a signal that is corrupted by noise, the model may overfit to the signal, ignoring the noise. In diffusion models, however, both the signal and the noise are faithfully represented, meaning that all variability in the data is taken into account. This is intuitive, given that the objective in training diffusion models is precisely to draw from a distribution with the same level of variability.

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

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

[2] 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.

[3] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models, December 2020. arXiv:2006.11239 [cs, stat].

[4] 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.

[5] Adam Block, Youssef Mroueh, and Alexander Rakhlin. Generative Modeling with Denoising Auto-Encoders and Langevin Sampling, October 2022. arXiv:2002.00107 [stat].

[6] Valentin De Bortoli. Convergence of denoising diffusion models under the manifold hypothesis, May 2023. arXiv:2208.05314 [stat].

[7] Xingchao Liu, Lemeng Wu, Mao Ye, and Qiang Liu. Let us Build Bridges: Understanding and Extending Diffusion Generative Models, August 2022. arXiv:2208.14699 [cs].

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

[9] Jakiw Pidstrigach. Score-Based Generative Models Detect Manifolds, October 2022. arXiv:2206.01018 [stat].

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

[11] Anand Jerry George, Rodrigo Veiga, and Nicolas Macris. Analysis of Diffusion Models for Manifold Data, February 2025. arXiv:2502.04339 [math].

[12] 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. arXiv:2505.17638 [cs].

[13] 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. Publisher: American Physical Society.

[14] Alessandro Favero, Antonio Sclocchi, and Matthieu Wyart. Bigger Isn't Always Memorizing: Early Stopping Overparameterized Diffusion Models, September 2025. arXiv:2505.16959 [cs].

[15] Alessandro Favero, Antonio Sclocchi, Francesco Cagnetta, Pascal Frossard, and Matthieu Wyart. How compositional generalization and creativity improve as diffusion models are trained, March 2025. arXiv:2502.12089 [stat].

[16] 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.

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

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

[19] 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. Publisher: American Physical Society.

[20] Florentin Guth, Zahra Kadkhodaie, and Eero P. Simoncelli. Learning normalized image densities via dual score matching, June 2025. arXiv:2506.05310 [cs].

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

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

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

[24] 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. Publisher: American Physical Society.

[25] Giulio Biroli and Marc Mézard. Generative diffusion in very large dimensions. *Journal of Statistical Mechanics: Theory and Experiment*, 2023(9):093402, September 2023.

[26] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation, May 2015. arXiv:1505.04597 [cs].

[27] 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. Publisher: American Physical Society.

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

[29] Binxu Wang and John J. Vastola. The Hidden Linear Structure in Score-Based Models and its Application, November 2023. arXiv:2311.10892 [cs].

[30] Xiang Li, Yixiang Dai, and Qing Qu. Understanding Generalizability of Diffusion Models Requires Rethinking the Hidden Gaussian Structure. November 2024.

[31] Pankaj Mehta and David J. Schwab. An exact mapping between the Variational Renormalization Group and Deep Learning, October 2014. arXiv:1410.3831 [stat].

[32] Serena Bradde and William Bialek. PCA Meets RG. *Journal of Statistical Physics*, 167(3):462–475, May 2017.

[33] Tanguy Marchand, Misaki Ozawa, Giulio Biroli, and Stéphane Mallat. Multiscale Data-Driven Energy Estimation and Generation. *Physical Review X*, 13(4):041038, November 2023.

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

[35] Etienne Lempereur and Stéphane Mallat. Hierarchic Flows to Estimate and Sample High-dimensional Probabilities, May 2024. arXiv:2405.03468 [stat].

[36] Andi Han, Wei Huang, Yuan Cao, and Difan Zou. On the Feature Learning in Diffusion Models, December 2024. arXiv:2412.01021 [stat].

A Diffusion models

Diffusion models are designed to reverse the noising process: they implement a mapping $\epsilon_\theta\left(x_t, t\right)$ whose paramaters θ are optimized to predict the noise vector ϵ_t. For a dataset $\mathcal{D}$ consisting of N samples in $\mathbb{R}^d$, this amounts to minimizing

$$L = \frac{1}{dT|\mathcal{D}|} \sum_t \sum_{x_0 \in \mathcal{D}} \mathbb{E}_{\epsilon_t} \left\| \epsilon_t - \epsilon_\theta\left(x_t\left(\epsilon_t\right), t\right) \right\|^2 , \tag{7}$$

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

$$\epsilon_\theta\left(x_t, t\right) = W_t \left(x_t - \sqrt{\bar{\alpha}_t} b_t\right) \tag{8}$$

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. (7) 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 γ_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

$$L = \frac{1}{dT|\mathcal{D}|} \sum_t \sum_{x_0 \in \mathcal{D}} \mathbb{E}_{\epsilon_t} \left\| \epsilon_t - W_t \left(\sqrt{\bar{\alpha}_t} x_t + \sqrt{1-\bar{\alpha}_t}\epsilon_t - \sqrt{\bar{\alpha}_t} b_t\right) \right\|^2 + \gamma_t \mathrm{Tr} W_t W_t^{\mathrm{T}} . \tag{9}$$

A.1 Optimizing the loss

Since W_t, b_t are not coupled in eq. (7) across different values of t, we can find their optima independently for each noising stage. Inserting eq. (8) into eq. (7), we find that the contribution for each t decomposes into a data-dependent term and one which depends only on the additive noise

$$\mathbb{E}_{\epsilon_t} \left\| \epsilon_t - \epsilon_\theta\left(x_t, t\right) \right\|^2 \;=\; \bar{\alpha}_t \left\| W_t \left(x_0 - b_t\right) \right\|^2 + \mathbb{E}_{\epsilon_t} \; \left\| \left(1 - \sqrt{1-\bar{\alpha}_t} W_t\right) \epsilon_t \right\|^2 . \tag{10}$$

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. (10) we take the average over infinitely many realizations of ϵ_t.

Eq. (10) 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 μ_0 and covariance Σ_0

$$\mu_0 := \frac{1}{N} \sum_{x_0 \in \mathcal{D}} x_0 , \qquad \Sigma_0 := \frac{1}{N} \sum_{x_0 \in \mathcal{D}} \left(x_0 - \mu_0\right) \left(x_0 - \mu_0\right)^{\mathrm{T}} =: \sum_\nu \lambda_\nu^0 e_\nu^0 \left(e_\nu^0\right)^T \tag{11}$$

of the training data. With this definitions, we now use that eq. (9) is quadratic in W_t, b_t, hence it is convex with the unique minimum

$$b_t^* = \mu_0 , \qquad W_t^* = \frac{\sqrt{1-\bar{\alpha}_t}}{\bar{\alpha}_t \Sigma_0 + \left(1 - \bar{\alpha}_t + \gamma_t\right) \mathrm{Id}} . \tag{12}$$

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

$$R = \sum_t \sum_\nu \left[\frac{\bar{\alpha}_t \lambda_\nu^0 + \gamma_t}{\left(\bar{\alpha}_t \lambda_\nu^0 + \left(1 - \bar{\alpha}_t + \gamma_t\right)\right)} \right] . \tag{13}$$

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

$$u_{s+1} = \mu_\theta(u_s, T - s) + \sigma_{T-s}\xi, \qquad \xi \sim \mathcal{N}(0, \mathrm{Id}) \,. \tag{14}$$

where we defined

$$\mu_\theta(x_t, t) := \frac{x_t - \sqrt{1 - \bar{\alpha}_t}\epsilon_\theta(x_t, t)}{\sqrt{\alpha_t}} + \sqrt{(1 - \sigma_t^2) - \bar{\alpha}_{t-1}}\epsilon_\theta(x_t, t) \,. \tag{15}$$

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 μ_0 and covariance Σ_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. (14). Before we do so, we note that to sample from a given Gaussian distribution with mean μ_0 and covariance Σ_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

$$u_T = \sqrt{\Sigma_0}u_0 + \mu_0 \,. \tag{16}$$

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. (14), 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

$$\hat{\beta}(t) := \frac{\beta_t}{h} \qquad\qquad \hat{\sigma}(t) := \frac{\sigma_t}{\sqrt{h}} \,. \tag{17}$$

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 $h \to 0$,

$$\frac{d\bar{\alpha}(t)}{dt} = -\hat{\beta}(t)\bar{\alpha}(t) \,. \tag{18}$$

This differential equation admits a formal solution, namely

$$\bar{\alpha}(t) = e^{-\zeta(t)}, \zeta(t) := \int_0^t ds\, \hat{\beta}(s) \,. \tag{19}$$

B.2 Fokker-Planck equation

With the definition of the continuous noising/denoising time limit in hand, we now write eq. (14) 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. (15) and eq. (12) into eq. (14) and Z_s is a Wiener process. We find

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

Observe that $m(s)$ is diagonal in the eigenbasis of Σ_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_ν fixed, dropping the index ν 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

$$\partial_s \rho(u,s) = -\partial_u\left[(m(s)u(s) + c(s))\,\rho(u,s)\right] + \frac{1}{2}\partial_u^2\left[\hat{\sigma}(1-s)^2\rho(u,s)\right]\,. \tag{20}$$

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

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

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

$$\partial_s(\sigma_u(s)^2) = 2m(s)\sigma_u(s)^2 + \hat{\sigma}(1-s)^2 \qquad \partial_s\mu_u(s) = m(s)\mu_u(s) + c(s). \tag{21}$$

which admit the solutions

$$\mu_u(s) = \exp\left(\int_0^s dv\, m(v)\right)\left[\int_0^s dv\, \exp\left(-\int_0^v dw\, m(w)\right)c(v) + \mu_u(0)\right] \tag{22}$$

$$\sigma_u(s)^2 = \exp\left(2\int_0^s dv\, m(v)\right)\left[\int_0^s dv\, \exp\left(-2\int_0^v dw\, m(w)\right)\hat{\sigma}(1-v)^2 + \sigma_u(0)^2\right] \tag{23}$$

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

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

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

$$\mu_u(s) = \sqrt{\bar{\alpha}(1-s)}\mu_0 - \sqrt{\bar{\alpha}(1)\frac{\bar{\alpha}(1-s)(\lambda^0-1)+1}{\bar{\alpha}(1)(\lambda^0-1)+1}}\mu_0 \tag{24}$$

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

Since $\bar{\alpha}(1)$ vanishes exponentially with ζ, 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. (23) 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. (23), 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 ζ, 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 $\gamma_t = c\sqrt{\bar{\alpha}_t}$, we must replace λ^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 τ and a learning rate η. At each training step, we will update the parameters of the linear network θ 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. (12)) we find, that in the limit $d\tau \to 0$

$$w_{\mu,\nu,t}(\tau) = w^*_{\mu,\nu,t} + \exp\left\{-2\frac{\eta}{dT}\left[\bar{\alpha}_t\lambda^0_\nu + (1-\bar{\alpha}_t+\gamma_t)\right]\tau\right\}\left(w_{\mu,\nu,t}(0) - w^*_{\mu,\nu,t}\right) . \tag{25}$$

This expression shows that through training, the entries of W_t approach their optimal value exponentially with rate $2\frac{\eta}{dT}\left[\bar{\alpha}_t\lambda^0_\nu + (1-\bar{\alpha}_t+\gamma_t)\right]$.

C.1 The speed of learning and overfitting

The elements of W_t exponentially relax towards 12 at a different rates $\bar{\alpha}_t\lambda^0_\nu+1-\bar{\alpha}_t+\gamma_t$ 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. (2), whose minimal values lead to the most severe overfitting. More precisely, the gap between training and test loss evolves with the training time for τ as

$$\frac{d\left(L(\tau) - R(\tau)\right)}{d\tau} = \frac{4\eta}{d^2T^2}\sum_t(1-\bar{\alpha}_t)\sum_\mu\left(1-e^{-\eta_{\mu,t}\tau}\right)\left(\frac{\bar{\alpha}_t\left(e^0_\mu\right)^{\mathrm{T}}\Sigma\, e^0_\mu + (1-\bar{\alpha}_t)}{\bar{\alpha}_t\lambda^0_\mu + (1-\bar{\alpha}_t+\gamma_t)} - 1\right)$$

In particular, starting from initially zero difference between these quantities, modes where $\left(e^0_\mu\right)^{\mathrm{T}}\Sigma\, e^0_\mu > \bar{\alpha}_t\lambda^0_\mu + \gamma_t$ will make the gap between training and test loss widen over time. This means that the most precarious directions in the sense of overfitting are also the ones which are learned the slowest. At the same time, regularization speeds up the learning process in all directions, as it increases the rate of convergence to the optimum. This makes early stopping an effective strategy to prevent overfitting.

D Replica theory for linear denoisers at finite N

In this appendix, we derive summary statistics for linear denoisers optimized using the empirical covariance matrix Σ_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 Σ to be

$$\Sigma = R\Lambda R^{\mathrm{T}}, \quad \Lambda = \begin{pmatrix}\lambda_1 & \dots & 0\\ \vdots & \ddots & \vdots\\ 0 & \dots & \lambda_d\end{pmatrix} \tag{26}$$

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

$$\begin{aligned} f_g(J) =& \frac{1}{d}\int\prod_\beta d\rho\left(x^\beta\right)\ln Z(J,\Sigma_0)\\ Z(J,\Sigma_0) :=& \left|\mathrm{Id} + \hat{\alpha}\Sigma_0 + RJg(\Lambda)R^{\mathrm{T}}\right|^{-\frac{1}{2}}\\ =& \int\frac{d\eta}{\sqrt{2\pi}^d}\exp\left(-\frac{1}{2}\eta^{\mathrm{T}}\left[\mathrm{Id} + \frac{\hat{\alpha}}{N}\sum_{\beta=1}^{N}\Sigma^{\frac{1}{2}}x^\beta\left(\Sigma^{\frac{1}{2}}x^\beta\right)^{\mathrm{T}} + RJg(\Lambda)R^{\mathrm{T}}\right]\eta\right) \end{aligned} \tag{27}$$

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.

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

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^β are coupled in the logarithm. To evaluate the integral, we now use the replica trick, which consists of re-writing the logarithm as

$$\langle \ln Z(J, \Sigma_0) \rangle_{\Sigma_0} = \lim_{n \to 0} \frac{1}{n} \left(\langle Z(J, \Sigma_0) \rangle_{\Sigma_0}^n - 1 \right) \tag{28}$$

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 η^α, where α is the replica index,

$$\langle Z(J, \Sigma_0) \rangle_{\Sigma_0}^n = \int \left(\prod_{\alpha=1}^n \frac{d\eta^\alpha}{\sqrt{2\pi}^d} \right) \left\langle \exp\left(\sum_{\alpha=1}^n -\frac{1}{2} \left(\eta^\alpha\right)^{\mathrm{T}} \left(\mathrm{Id} + \hat{\alpha}_t \Sigma_0 + R J R^{\mathrm{T}} f(\Sigma) \right) \eta^\alpha \right) \right\rangle_{\Sigma_0} \tag{29}$$

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

D.1 Introducing auxiliary variables

We first isolate the terms depending on x^β from the expression.

$$\begin{aligned} \langle Z \rangle_{\Sigma_0}^n = \int \left(\prod_{\alpha=1}^n \frac{d\eta^\alpha}{\sqrt{2\pi}^d} \right) \exp\left(\sum_{\alpha=1}^n \left[-\frac{1}{2} \left(\eta^\alpha\right)^{\mathrm{T}} \left(\mathrm{Id} + R J f(\Lambda) R^{\mathrm{T}} \right) \right] \right) \\ \cdot \int \left(\prod_{\beta=1}^N \frac{dx^\beta}{\sqrt{2\pi}^d} \right) \exp\left(-\frac{1}{2} \sum_\beta \left(x^\beta\right)^{\mathrm{T}} \left(\frac{\hat{\alpha}}{N} \sum_\alpha \Sigma^{\frac{1}{2}} \eta^\alpha \left(\Sigma^{\frac{1}{2}} \eta^\alpha \right)^{\mathrm{T}} + \mathrm{Id} \right) x^\beta \right) \end{aligned} \tag{30}$$

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 Σ appears, but not the samples x^β, and vice versa. Additionally, note that for given μ^α, the second line is just the $N-$th power of the first line. Both factors are coupled together by the fact that μ^α 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^β and μ^α by R^{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 μ^α decouple. To this end, we define our first auxiliary variable

$$R_\alpha = \frac{1}{\sqrt{d}} x^{\mathrm{T}} \mu^\alpha \tag{31}$$

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

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

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:

$$Q_{\alpha_1,\alpha_2} := \frac{1}{d} \left(\mu^{\alpha_1}\right)^{\mathrm{T}} \mu^{\alpha_2} \tag{32}$$

Using the Hubbard-Strantonivic transform backwards to also eliminate the integrals over all R_α, 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

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

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

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

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

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

$$\begin{aligned} \bar{Z}^n(j) &= \int \prod_{\alpha_1 \le \alpha_2}^n \frac{dQ_{\alpha_1,\alpha_2} d\tilde{Q}_{\alpha_1,\alpha_2}}{2\pi} \exp\left(dS\left(Q, \tilde{Q}\right) \right) \\ S\left(Q, \tilde{Q}\right) &= \left[i \sum_{\alpha_1 \le \alpha_2} \frac{1}{d} \tilde{Q}_{\alpha_1,\alpha_2} Q_{\alpha_1,\alpha_2} + \frac{1}{d} \sum_i S_i(\tilde{Q}) + \frac{N}{d} \ln G(Q) \right] \end{aligned} \tag{33}$$

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.

D.2 Saddle point approximation for any n

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

$$Q_{\alpha_1,\alpha_2} = q\delta_{\alpha_1,\alpha_2} + p(1 - \delta_{\alpha_1,\alpha_2}) \tag{34}$$

$$\frac{i}{d}\tilde{Q} = \tilde{q}\delta_{\alpha_1,\alpha_2} + \tilde{p}(1 - \delta_{\alpha_1,\alpha_2})\,. \tag{35}$$

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

$$S_i(\tilde{Q}) = -\frac{n-1}{2}\ln\left|1 + J_i g(\lambda_i) + \lambda_i\left(2\tilde{q} - \tilde{p}\right)\right| - \frac{1}{2}\ln\left|1 + J_i g(\lambda_i) + \lambda_i\left(2\tilde{q} + (n-1)\tilde{p}\right)\right| - n\ln\lambda_i$$

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:

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

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

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

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

$$q(J) = \frac{1}{d}\sum_i \left[\frac{\lambda_i}{1 + J_i g(\lambda_i) + \lambda_i\left(\frac{N}{d}\frac{1}{q + \frac{N}{d\hat{\alpha}}}\right)}\right] \tag{36}$$

and

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

D.3 Taking the limit $n \to 0$

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

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

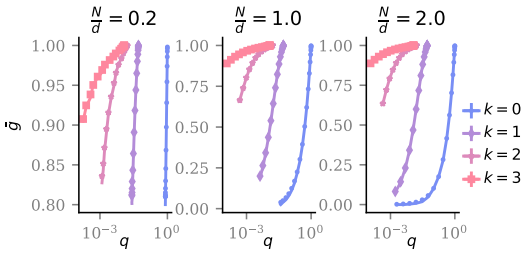

Figure 3: Relation between $\bar{g}$ and q. Full lines are predictions using eq. (43), markers show simulations for different fractions of N/d, and varying values of $\hat{\alpha}$ between $2^{-9}, 2^{9}$. Colors and marker styles distinguish and different spectra of the true covariance matrix Σ, namely $\lambda_i = i^{-k}\, \forall i = 1, \ldots, d$.

Due to the linear appearance of n in $\ln \bar{Z}^n(J)$, we can finally take the limit $n \to 0$ in eq. (28) and find the expression for f

$$f(J) = \frac{N}{2d}\frac{q(J)}{q(J)+\frac{N}{d\hat{\alpha}}} - \left[\frac{1}{2d}\sum_i \ln\left|1 + J_i g(\lambda_i) + \lambda_i\left(\frac{N}{d}\frac{1}{q(J)+\frac{N}{d\hat{\alpha}}}\right)\right|\right] - \frac{N}{2d}\ln\left(\frac{d\hat{\alpha}}{N}q(J)+1\right). \tag{37}$$

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

$$-2\sum_i \frac{df}{dJ_i}\bigg|_{J=0,g(x)=x} = \frac{1}{d}\left\langle \mathrm{Tr}\Sigma\frac{1}{\mathrm{Id}+\hat{\alpha}\Sigma_0}\right\rangle_{\Sigma_0} = \frac{1}{d}\sum_i \frac{\lambda_i}{1+\lambda_i\left(\frac{N}{d}\frac{1}{q+\frac{N}{d\hat{\alpha}}}\right)} = q \tag{38}$$

where the first equality follows from the definition of f, see eq. (27), the second follows from eq. (37) and the final equality is a consequence of eq. (36).

D.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:

$$g(z) = \int d\rho_\Sigma(\lambda)\,\frac{1}{z - \lambda\left(1-\frac{d}{N}+\frac{d}{N}zg(z)\right)} \tag{39}$$

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

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

this variable then fulfills the self-consistency equation

$$\bar{g} = \frac{1}{d}\sum_i \frac{1}{1+\hat{\alpha}\lambda_i\left(1-\frac{d}{N}+\frac{d}{N}\bar{g}\right)}. \tag{40}$$

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. (36), we find

$$s(q) := \frac{\hat{\alpha}\left(\frac{d}{N} - 1\right) q + 1}{\hat{\alpha}\frac{d}{N} q + 1}\,, \tag{41}$$

Furthermore, it follows from the definition eq. (41), 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 $s(q)$,

$$s(q) = \frac{1}{d} \sum_i \frac{1}{1 + \hat{\alpha}\lambda_i \left(1 - \frac{d}{N} + \frac{d}{N} s(q)\right)} \tag{42}$$

which is identical to the self-consistency equation for $\bar{g}$, eq. (40), meaning that the replica calculation produces a variant of the known self-consistency relation eq. (39) 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 Σ, namely

$$\bar{g} = s(q) \tag{43}$$

We test this relation in fig. 3, finding excellent agreement with simulations.

D.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

$$\psi_{1,1}^t = \left\langle \mathrm{Tr}\left[\frac{1 - \bar{\alpha}_t}{\left(\mathrm{Id} + \hat{\alpha}_t \Sigma_0\right)^2} \right] \right\rangle \quad \psi_{1,2}^t = \left\langle \mathrm{Tr}\left[\frac{\bar{\alpha}_t}{\left(\mathrm{Id} + \hat{\alpha}_t \Sigma_0\right)^2} \Sigma \right] \right\rangle \quad \psi_2^t = \left\langle \mathrm{Tr}\left[\frac{1}{\left(\mathrm{Id} + \hat{\alpha}_t \Sigma_0\right)} \right] \right\rangle \tag{44}$$

We first compute ψ_2, using that

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

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. (40).

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

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

Where we used Wick's theorem to evaluate the Gaussian moments. For the specific functions of interest eq. (44), 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

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

Using the self-consistency equation, we find

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

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

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

D.6 Residual and test loss at finite N

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

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

Inserting this into the equation for the residual loss, eq. (13), which yields

$$\begin{aligned}R &= \frac{-2}{T}\sum_t \frac{\bar{\alpha}_t}{1-\bar{\alpha}_t+\gamma_t}\left.\frac{df}{d\hat{\alpha}_t}\right|_{J=0,g=1} + \frac{\gamma_t}{1-\bar{\alpha}_t+\gamma_t}\psi_2 \\ &= \frac{1}{T}\sum_t \frac{\bar{\alpha}_t}{1-\bar{\alpha}_t+\gamma_t}\frac{q}{\frac{d\hat{\alpha}_t}{N}q+1} + \frac{\gamma_t}{1-\bar{\alpha}_t+\gamma_t}\frac{1}{d}\sum_i\frac{1}{1+\lambda_i\left(\frac{N}{d}\frac{1}{q+\frac{N}{d\hat{\alpha}_t}}\right)}\end{aligned} \tag{45}$$

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 D.5.

D.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

$$\mathrm{DKL}(\rho_N|\rho) = \frac{1}{2}\left[\ln\frac{|\Sigma|}{|\Sigma_0+c\mathrm{Id}|} + (\mu-\mu_0)^{\mathrm{T}}\Sigma^{-1}(\mu-\mu_0) + \mathrm{Tr}\Sigma^{-1}\left(\Sigma_0+c\mathrm{Id}\right) - d\right]\,, \tag{46}$$

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

$$\mathrm{Tr}\Sigma^{-1}\left(\langle\Sigma_0\rangle + c\mathrm{Id}\right) - d = c\,\mathrm{Tr}\Sigma^{-1} \tag{47}$$

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 $c=\frac{1}{\hat{\alpha}}$

$$-\frac{1}{2}\left\langle\ln\left|\Sigma_0+c\mathrm{Id}\right|\right\rangle = d\,f(J=0)+\frac{d}{2}\ln\hat{\alpha} \tag{48}$$

we now evaluate eq. (37) 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. (5).

D.8 Bounds and approximations of q

D.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

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

First, note that from eq. (38) 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

$$d<N \quad \Rightarrow \quad q \leq \frac{1}{\hat{\alpha}\left(1-\frac{d}{N}\right)} \tag{49}$$

For $\hat{\alpha}$ very large, one can show that the difference to the right hand side (see appendix D.8.2) 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

$$q \leq \frac{1}{d}\mathrm{Tr}\Sigma \tag{50}$$

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

$$q_0 = \frac{1}{d}\mathrm{Tr}\Sigma, \quad q_{n+1} = \left(\frac{d}{N}q_n+\frac{1}{\hat{\alpha}}\right)\frac{1}{d}\mathrm{Tr}\frac{\Sigma}{\Sigma+\mathrm{Id}\left(\frac{d}{N}q_n+\frac{1}{\hat{\alpha}}\right)} \tag{51}$$

we then find that $q \leq q_n \forall n$. We compare numerical simulations of eq. (38) to the bound on q obtained thus in fig. 4, finding that the bound becomes increasingly tight as we increase n.

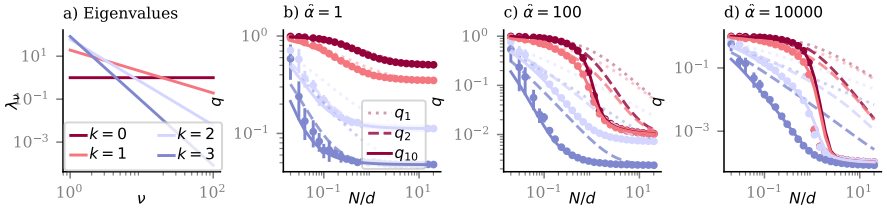

Figure 4: Comparison of q to the upper bound. a) shows the spectra of the covariances we consider, $\lambda_\nu = c_k \nu^{-k}$ where we choose c_k such that $\frac{1}{d}\sum_\nu \lambda_\nu = 1$. b)-f) compare numerical values of q found using eq. (38) with $d = 100$. Dots and error bars denote mean and standard deviations over ten realizations of Σ_0, respectively. Dotted, dashed and full lines show upper bounds eq. (51) for increasing n.

D.8.2 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. (43). We now additionally assume that $\hat{\alpha}$ is much larger than $\left(\lambda^0_{\text{min}}\right)^{-1}$, where λ^0_{min} is the smallest eigenvalue in Σ_0. This assumption is only valid for at least $N \geq d$ as otherwise Σ_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

$$q = \frac{1}{\hat{\alpha}\frac{d}{N}}\left(\frac{1}{1-\frac{d}{N}+\frac{d}{N}\bar{g}} - 1\right) \approx \frac{1}{\hat{\alpha}\frac{d}{N}}\left(\frac{1}{1-\frac{d}{N}} - 1\right) \tag{52}$$

which is independent of Σ. Inserting this into eq. (5), we find

$$\frac{\mathrm{DKL}(\rho_N|\rho)}{d} = \frac{1 + \frac{N}{d}\ln(1-\frac{d}{N}) + \ln(1-\frac{d}{N}) + \frac{N}{d^2}}{2} + \mathcal{O}\left(\sqrt{\hat{\alpha}^{-1}}\right), \tag{53}$$

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

E 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 ~ 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. 5 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 ν 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 $N \geq 6400$).

F Differences between linear and non-linear models

We now test whether the mappings encoded by different architectures are indeed similar. Prior studies have observed increasing similarity between non-linear and linear models with t, see [29, 30]. We compute the relative distance of their mappings in the eigen-space of Σ. We define a direction - and t

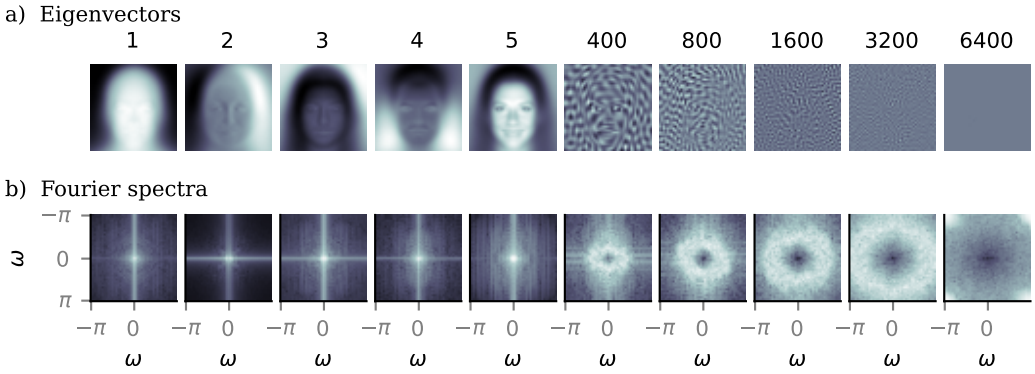

Figure 5: a) First five leading eigenvectors $\nu = 1, \ldots, 5$ of the covariance matrix of the CelebA data, as well as sub-leading eigenvectors $\nu = 400, \ldots, 6400$. b) Corresponding Fourier spectra of the eigenvectors.

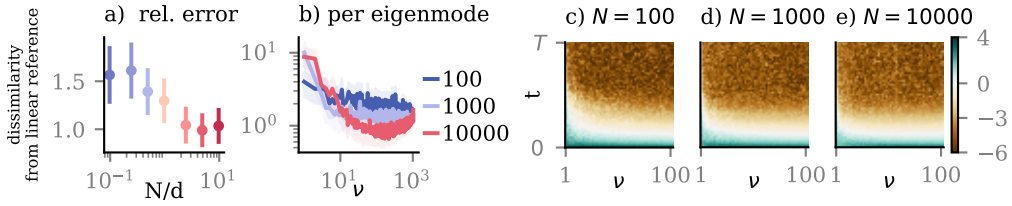

Figure 6: Relative difference of non-linear denoisers from best linear model, per noising step t and direction ν, trained on increasing numbers of data. a) averaged over ν and t, b) averaged over t, c) - e) $\log d_{t,\nu}$ per ν, t. All data are averaged over 10^2 test samples from CIFAR-10 per t, ν.

dependent distance $d_{t,\nu}$ measure

$$d_{t,\nu} = \frac{\left(\epsilon^N\left(x_t, t\right) - \epsilon^*_\infty\left(x_t, t\right)\right)^2_\nu}{\left|\left(\epsilon^N\left(x_t, t\right) + \eta\right)_\nu \left(\epsilon^*_\infty\left(x_t, t\right) + \eta\right)_\nu\right|} \tag{54}$$

where ϵ^N is a U-net trained on N examples and ϵ^*_∞ 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 fig. 7 Overall, we find that the relative error decays with N, ν and t. Indeed for a large extent of t, ν, the difference between linear and non-linear models becomes very small. However, for leading eigenmodes (small ν), the differences between linear and non-linear models grow with N. This (small ν, small t) is also the regime where we expect the non-Gaussianity of the data to have the largest effect.

In fig. 7 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, ν and t. Indeed for a large extent of t, ν, the difference between linear and non-linear models becomes very small. However, for leading eigenmodes (small ν), the differences between linear and non-linear models grow with N.

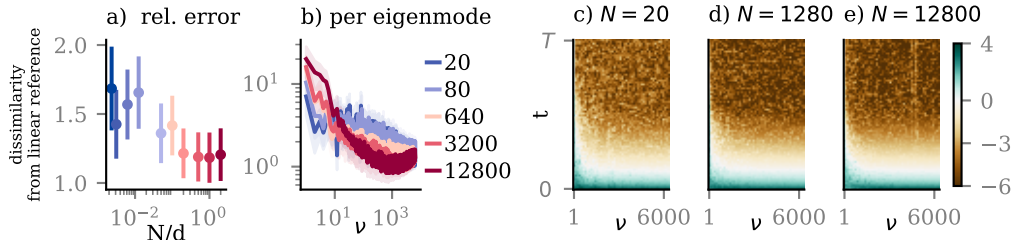

Figure 7: Relative difference of non-linear denoisers from best linear model, per noising step t and direction ν, trained on inecreasing numbers of data. a) averaged over ν and t, b) averaged over t, c) - e) $\log d_{t,\nu}$ per ν, t. All data are averaged over 100 test samples per t, ν.