Local Coverage Governs Memorization in Diffusion Models

Anonymous authors
Paper under double-blind review

Abstract

Memorization in diffusion models is often treated as a global property of the model or dataset. In practice, however, a single diffusion model can simultaneously generate both memorized and novel samples. Which training samples are most likely to be memorized? In this work, we show that memorization is governed by local data coverage. Leveraging the connection between diffusion models and kernel density estimation (KDE), we derive a theoretical criterion that predicts whether a point is memorized based on the density of training data in its neighborhood and the size of the training dataset. In the high-dimensional limit, this leads to a sharp, local transition: regions of low coverage are dominated by isolated training samples, which are memorized, while dense regions support interpolation and generalization. We validate these predictions empirically, showing that memorization increases with local sparsity and that diffusion models exhibit a coexistence of memorized and novel samples within the same model. Extending this framework to multi-class settings, we further show that classes with higher intra-class diversity (and thus lower local coverage) are more strongly memorized. Our results provide a local view of memorization in diffusion models, explaining when and where memorization occurs in terms of data geometry.

1 Introduction

How does memorization arise locally in diffusion models? Diffusion models are powerful generative methods capable of learning complex high-dimensional data distributions from finite datasets. Yet when trained with limited data, they may reproduce training examples rather than generate novel samples, a phenomenon commonly referred to as memorization (Somepalli et al., 2022; Carlini et al., 2023; Kadkhodaie et al., 2024). Understanding the mechanisms that drive memorization is important because memorized samples can lead to unintended data leakage, privacy violations, and the release of copyrighted or sensitive training data.

Theoretical work has begun to analyze memorization in linear models (Maillard & Goldt, 2026; Wang et al., 2026), in random feature models (George et al., 2026; Bonnaire et al., 2026), and by exploiting the close connection between diffusion models and kernel density estimation (KDE) (Pidstrigach, 2022; Pham et al., 2024; Ambrogioni, 2024; Biroli & Mézard, 2024; Achilli et al., 2025a; Li et al., 2024; Lucibello & Mézard, 2024). In the latter picture, each training sample contributes a local kernel, and generalisation arises from the superposition of many such kernels, see fig. 1a for a sketch. When kernels overlap strongly, the model assigns finite probability to the space between examples; when overlap is weak, generation can collapse onto individual training points. These analyses predict a global phase transition: the whole generative model goes from memorizing to generating new samples at once.

Recent work has begun to paint a more fine-grained picture of memorization. In practice, it was observed that models often reproduce only a subset of their training data rather than the entire dataset (Fang et al., 2025; Carlini et al., 2023); the effect depends strongly on the training setup (see further related work below). Thus, at finite sample size, a diffusion model often generates a mixture of copied and novel samples see fig. 1 c) f) and i). On the theoretical side, Achilli et al. (2026) showed that memorization may emerge progressively

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

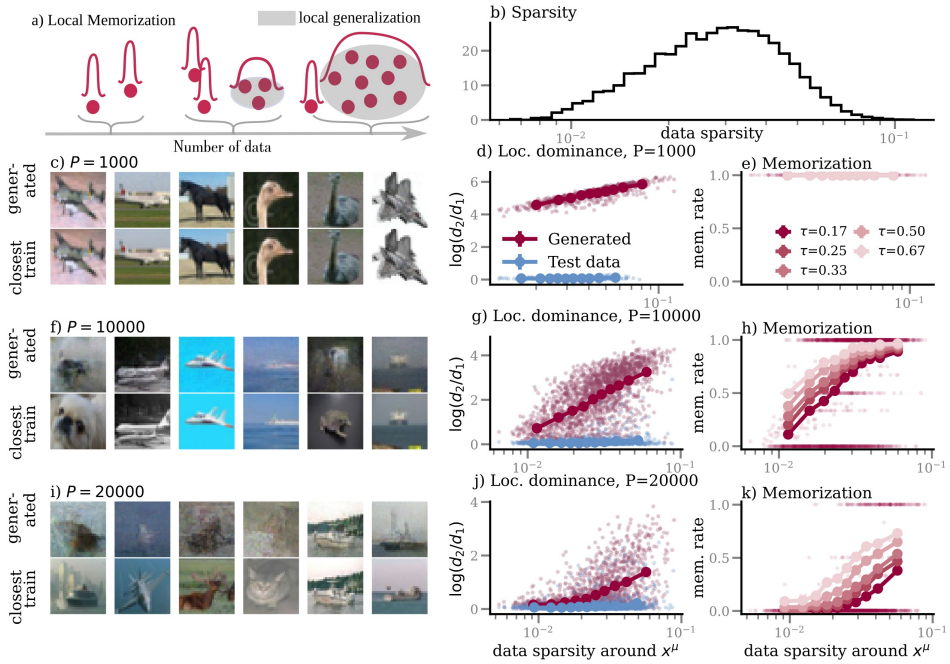


Figure 1: Local data coverage governs memorization. a) Sketch of local memorization in kernel density approximation. b) Histogram of data sparsity, measured by the average squared distance of a training point its 10 nearest neighbors, for 20,000 CIFAR10 samples. c) Generated samples + closest training image (measured by ℓ_2 distance) for a diffusion model trained on $P = 1000$ training examples. d) To measure how closely a training data point x^μ is replicated given its local sparsity, we generate $40,000$ training data. For each generated sample we compute the local dominance as the log-ratio $\log d_2/d_1$ between the distances d_1 and d_2 to its nearest and second-nearest training samples. We then aggregate these data by training sample To account for spurious correlations with local dominance, we compare against a baseline computed from test data (blue). Scatter shows raw values, markers indicate binned averages. e) Memorization rate per training sample, defined as the fraction of generated samples satisfying $d_1/d_2 < \tau$. Scatter shows raw values for $\tau = 0.17$, markers indicate binned averages across thresholds. f)–k) Same measurements for models trained on larger datasets. As the number of training samples increases, both dominance and memorization decrease, and their dependence on sparsity weakens. Overall, sparse regions exhibit strong single-sample dominance and higher memorization, while dense regions promote interpolation across multiple training points.

through the loss of manifold dimensions, leading to a form of geometric memorization in which some directions of variability are lost before exact copying occurs. Garnier-Brun et al. (2026) showed a coexistence of memorization and generalization in early-stopped diffusion models. Zhang et al. (2025) showed in a two-layer ReLU denoising autoencoder that locally sparse data admit solutions storing individual examples in the weights, whereas locally abundant data yield solutions encoding shared statistics.

These results establish that memorization and generalization can coexist, but do not provide a general criterion that predicts the memorization risk of a given sample. We therefore ask: Which regions, classes, or individual samples are most likely to be memorized?

In this work, we give a concrete criterion for memorization by introducing a local theory of memorization in diffusion models. Our central hypothesis is simple: memorization is governed not only by the total number of training samples, but by their local coverage. Regions of data space containing many nearby examples support interpolation and generalization, whereas isolated regions are prone to sample retrieval. In fig. 1 we show an example of this behavior for diffusion models with U-net architecture (Ronneberger et al., 2015) trained

on subsets of the CIFAR10 image dataset (Krizhevsky, 2009) of increasing size P. When trained on very few data points, the models first produce only memorized data. As we increase the number of training data, the models begin to generate a mixture of memorized and novel examples (panels c,f,i). We then quantify to which degree samples generated from diffusion models replicate the closest training data (panels d,g,j) as a function of local sparsity. We find isolated training points in low density (high sparsity) regions are preferentially memorized (panels e,h,k), whereas samples generated in regions of higher local density (lower local sparsity in fig. 1) are less likely to be memorized. See section 4 for full experimental details.

To formalize this intuition, we develop a local theory of memorization using the KDE framework of Lucibello & Mézard (2024); Biroli & Mézard (2024); Achilli et al. (2025b). Here, we sketch the main idea. We quantify the local coverage around a point $x \in \mathbb{R}^N$ by the probability mass contained in a ball of radius R around x,

$$p_{\text{in},R}(x) = \int_{B_R(x)} d\rho(x'), \qquad \nu_{\text{in},R}(x) = \lim_{N\to\infty} \tfrac{1}{N} \ln p_{\text{in}}(x)\,. \tag{1}$$

where $B_R(x)$ denotes a N-dimensional ball centered at x and ρ the density of the data from which the training points are drawn. This quantity measures how densely the data distribution ρ populates the neighborhood of x. In the high-dimensional regime in which the number of samples scales as (Lucibello & Mézard, 2024; Biroli & Mézard, 2024) $P = e^{\alpha N}$. Our main result shows that memorization at a point x is controlled by the pair $(\nu_{\text{in}}(x), \alpha)$ with a sharp transition:

$$\begin{cases} x \in A_{\text{mem}} & \text{if } \nu_{\text{in},R}(x) < -\alpha \\ x \in A_{\text{gen}} & \text{otherwise} \end{cases} \tag{2}$$

where we refer to A_{gen} as the region where the model correctly interpolates, and A_{mem} is a region where the learned density is characterized by isolated peaks centered on training examples. This result shows that depending on local coverage, the same diffusion model may exhibit retrieval-like behavior in some regions of space and generative interpolation in others. Points in low-coverage regions, where $\nu_{\text{in},R}(x)$ is small, are memorized, while points in high-coverage regions are generalized. We give a more detailed account of related work in section A.

Our main contributions are then as follows

- We develop a local theory of memorization in KDE and prove that the transition from memorization to generalization is governed by the local kernel statistic ν_{in}.
- We show that the local KDE theory explains several empirical properties of diffusion models:
  - Coexistence of memorization and generalization: the same model can simultaneously generate copied and novel samples; as P increases, memorization vanishes first in dense regions and persists longest in sparse regions.
  - Class-dependent memorization: classes with larger intra-class diversity (and therefore lower local coverage) are memorized more strongly
- Finally, we show empirically that local coverage also predicts which data are memorized first during diffusion model training.

More broadly, our results suggest that memorization in diffusion models is not a global binary property, but a local finite-sample phenomenon governed by the geometry and coverage of the training distribution. This mechanistic insight into memorization opens up the possibility of designing dedicated strategies to both suppress unwanted memorization and to retain specific training samples.

2 Background: from diffusion models to kernel density estimation

We briefly recall a key observation: diffusion models trained on finite data behave like kernel density estimators (KDE). Diffusion models define a forward noising process that gradually corrupts a data point x_0 from ρ through the addition of noise

$$x_t(\epsilon_t) = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon_t, \qquad \epsilon_t \sim \mathcal{N}(0, \text{Id})\,, \tag{3}$$

where $\bar{\alpha}_t \in (0,1)$ 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 $\mathcal{N}(0, \mathrm{Id})$. A neural network ϵ_θ with parameters θ is trained to predict the noise ϵ_t, which corresponds to learning the score $\nabla \ln \rho_t(x_t)$, where ρ_t is the distribution of x_t. This is achieved via the denoising score matching objective

$$L = \frac{1}{NTP} \sum_t \sum_{x^\mu \in \mathcal{D}} \mathbb{E}_{\epsilon_t} \left\| \epsilon_t - \epsilon_\theta \left(x_t^\mu \left(\epsilon_t \right), t \right) \right\|^2 , \tag{4}$$

where $\mathcal{D}$ is a training set of size P points $\{x^\mu\}_{\mu=1}^P$ drawn i.i.d. from ρ.

To make the connection to KDE explicit, consider an idealized setting in which the model has infinite capacity and is trained to optimality. In this case, for each t, the learned function $\epsilon_\theta(\cdot, t)$ can be treated as an arbitrary function $\epsilon(\cdot, t)$ and the loss can be minimized functionally. Setting this functional derivative to zero yields:

$$\epsilon^*(x,t) = -\sqrt{1-\bar{\alpha}_t} \nabla_x \ln \sum_\mu \mathcal{N}\left(x; \sqrt{\bar{\alpha}_t} x^\mu, (1-\bar{\alpha}_t)\mathrm{Id}\right) . \tag{5}$$

This identity reveals that, when trained to convergence on a finite number of samples, diffusion models learn a Gaussian mixture centered at the training data points. In other words, diffusion models trained on a finite dataset recover the score of the empirical distribution convolved with Gaussian noise, rather than the true underlying density ρ. In the low-noise limit ($\bar{\alpha}_t \to 1$), this mixture becomes sharply peaked around individual training samples, and the model effectively reproduces them. At finite noise, there is a direct correspondence between eq. (5) and KDE. KDE attempts to approximate a density ρ using a mixture of kernel functions K_N, indexed by the input dimension N, centered at the data points:

$$Z_N(x) = \frac{1}{P} \sum_\mu K_N \left(\frac{||x - x^\mu||}{R} \right) . \tag{6}$$

The correspondence with KDE becomes explicit by identifying the kernel as Gaussian with bandwidth $R = \sqrt{1-\bar{\alpha}_t}$ up to a rescaling of the variables x, x^μ by $\sqrt{\bar{\alpha}_t}$. In the ideal case, one has sufficient number of samples P such that Z becomes a smooth approximation of ρ. This behavior with P is one hypothesis to explain memorization and generalization in diffusion models.

This perspective suggests that generalization is not uniform across the data space: even when the total number of samples is large, the quality of the KDE approximation depends on local sample density. Regions with high data density are well approximated, while sparse regions remain dominated by individual kernels. In the following, we formalize this intuition by deriving conditions under which local regions exhibit memorization or generalization.

3 Theory: local kernel density estimation

3.1 General theory and main result

To study kernel density estimation (KDE), we characterize the behavior of the log-density

$$z_N(x_N) = \frac{1}{N} \ln Z_N(x_N) . \tag{7}$$

where $Z_N(x)$ is the mixture of kernel functions K centered at the data points defined in eq. (6). The difficulty in establishing the behavior of z_N is that it is a random quantity that depends on the draw of the P training points from ρ. Our goal is to compute its distribution over such draws to derive general properties of high dimensional KDE.

Since the ambient dimension changes with N, the symbol x_N should not be understood as a fixed point in a fixed space. We instead consider a sequence of vectors $\mathbf{x} = (x_N)_{N \geq 1}$ and distributions ρ_N with $x_N \in \mathbb{R}^N$, and assume that the corresponding local statistics have a limit. Equivalently, the result is indexed not by the coordinate vector x_N itself, but by the limiting local environment around it, namely the distribution of

$$\epsilon_{\mathbf{x},R,N}(\tilde{x}_N) = \frac{1}{N} \ln K_N\left(\tfrac{x_N - \tilde{x}_N}{R}\right) , \qquad \tilde{x}_N \sim \rho_N ,$$

Thus, the behavior of $z_N(x_N)$ is controlled by the distribution of the scalar statistic $\epsilon_{\mathbf{x},R,N}$ rather than by the full high-dimensional variable $\tilde{x}_N \sim \rho_N$.

We use an exponential transform, inspired by Dotsenko (2011), to compute the cumulative distribution function of $z_N(x_N)$. Define

$$\overline{W_N(y, x_N)} = \overline{\exp\left(-\exp\left[-N\left(y - z_N(x_N)\right)\right]\right)}, \tag{8}$$

where the overline denotes the average over draws of the P training samples. As $N \to \infty$, the inner function converges pointwise to $e^{-e^{-N(\cdot)}} \to \Theta(\cdot)$, with convention $\Theta(0) = e^{-1}$. In this limit therefore, the exponential transform characterizes the probability that $z_N \leq y$. Using independence of the training samples, the average factorizes and gives

$$\overline{W_N(y, x_N)} = e^{e^{\alpha N} \ln[1 - I_N(y, x_N)]}, \quad \text{with} \quad I_{R,N}(y, x_N) = \mathbb{E}_{\tilde{x} \sim \rho_N}\left[1 - e^{-e^{N\left[\epsilon_{x,R,N}(\tilde{x}) - \alpha - y\right]}}\right]. \tag{9}$$

This yields the following theorem, which is our main theoretical result:

Theorem 1 (Cumulative distribution function of KDE). Assume that the cumulative distribution function converges $F_{\mathbf{x},N}(y) = \Pr(z_N(x_N) \leq y) \to F_{\mathbf{x}}(y)$ for all y as $N \to \infty$. Furthermore, let us assume that the limit

$$i_{R,\mathbf{x}}(y) := \lim_{N \to \infty} e^{\alpha N} I_{R,N}(y, x_N) \tag{10}$$

exists with $i_{R,\mathbf{x}}(y) \in [0, \infty]$. Then at every continuity point y of $F_{\mathbf{x}}$, we find

$$F_{\mathbf{x}}(y) = \begin{cases} 0 & \text{if } i_{R,\mathbf{x}}(y) = \infty, \\ \exp[-i_{R,\mathbf{x}}(y)] & \text{if } 0 < i_{R,\mathbf{x}}(y) < \infty, \\ 1 & \text{if } i_{R,\mathbf{x}}(y) = 0. \end{cases} \tag{11}$$

We detail the full proof in section B. Theorem 1 shows that the limiting CDF of $z_N(x_N)$ is determined by the exponential scaling of the probability that the kernel statistic $\epsilon_{\mathbf{x},N}$ is large enough to contribute at level y. In typical high-dimensional settings, $z_N(x_N)$ concentrates, so the intermediate regime in eq. (11) collapses to the location of a sharp jump. Fluctuations of $\epsilon_{\mathbf{x}}$ would instead broaden this transition.

3.2 Case study: hard-sphere kernels

We now specialize to the hard-sphere kernel

$$K_N(u) = \Theta(R^2 - \|u\|^2)/V_R, \tag{12}$$

where V_R is the volume of the N-dimensional ball of radius R. In high dimensions, the hard-sphere kernel captures the same radial concentration mechanism as a Gaussian kernel. At the level of the exponential transform, the relevant quantity is the distribution of $\epsilon_{x,R,N}(\tilde{x}_N)$. For a Gaussian kernel, this statistic is, up to additive constants, proportional to $-\frac{1}{N}\|x_N - \tilde{x}_N\|^2$. Thus the leading exponential behavior is governed by level sets of the squared distance. In high dimensions, the Gaussian measure concentrates on a thin shell at its typical radius. Similarly, for the hard-sphere kernel, almost all of the volume of the ball concentrates near its boundary. Therefore, while Gaussian and hard-sphere kernels are quantitatively different, the hard-sphere kernel provides a tractable model that exposes the same local-coverage transition mechanism.

In this case, we find that the quantity controlling the limiting behavior is the local coverage. Let us assume that the limit of the local coverage rate

$$\nu_{\text{in},R,\mathbf{x}} = \lim_{N \to \infty} \tfrac{1}{N} \ln p_{\text{in},R,N}(x_N). \tag{13}$$

exists. Then we obtain the following result:

Lemma 2 (Local hard-sphere KDE transition). Let $P = e^{\alpha N}$ with fixed $\alpha > 0$, and let K_N be the hard-sphere kernel in eq. (12) and $v_R = \lim_{N \to \infty} 1/N \ln V_R$. Under the assumptions of theorem 1 and additionally assuming that $\nu_{\text{in},\mathbf{x}}$ exists, when $\nu_{\text{in},\mathbf{x}} \neq -\alpha$ we find

$$z_N(x_N) \xrightarrow[N \to \infty]{\mathbb{P}} \begin{cases} \nu_{\text{in},R,\mathbf{x}} - v_R & \text{if } \nu_{\text{in},R,\mathbf{x}} > -\alpha, \\ -\infty & \text{if } \nu_{\text{in},R,\mathbf{x}} < -\alpha. \end{cases} \tag{14}$$

The proof follows from a direct application of theorem 1, see section B.

Lemma 2 establishes a local transition controlled by the comparison between local coverage $\nu_{\text{in},R}(x)$ and sample complexity α. Regions with $\nu_{\text{in},R}(x) > -\alpha$ contain exponentially many nearby samples and therefore support smooth KDE estimates. Regions with $\nu_{\text{in},R}(x) < -\alpha$ contain no nearby samples with high probability; in the diffusion-model interpretation, probability mass assigned to such regions concentrates on isolated training examples, leading to memorization. The divergence $z(x) \to -\infty$ for $\nu_{\text{in},R}(x) < -\alpha$ reflects the fact that, in these regions, the kernel density estimate vanishes with high probability. Indeed,

$$\Pr\left(\{Z(x) = 0\}\right) = (1 - p_{\text{in}}(x))^P = e^{P \ln(1 - e^{N\nu_{\text{in},R,N}(x)})} \to \Theta\left(-\left[\alpha + \nu_{\text{in},\mathbf{x}}\right]\right) .$$

On the other hand, when α is large enough, then $Z(x)$ approaches $p_{\text{in}}(x)$, hence the kernel density estimate converges to a local average of ρ over a sphere.

To see how these two regions correspond to memorization, we now make a classical leave-one out argument. Suppose we fix a sample x^μ. We then find that the estimated density at x^μ is given by

$$Z_{\mathcal{D}}(x^\mu) = \frac{1}{PV_R}\left[1 + \sum_{\nu \neq \mu} \mathbf{1}\{x^\nu \in B_R(x^\mu)\}\right]. \tag{15}$$

The first term is the self-kernel of x^μ, the second term contains the contributions of all other samples. If x^μ lies in the region where $\nu_{\text{in},R}(x^\mu) < -\alpha$, then the latter sum is zero in the high dimensional limit. Thus only the self-kernel contributes there, leading to memorization. On the other hand, if $\nu_{\text{in},R}(x^\mu) > -\alpha$, then an exponential number of other samples contributes to the latter sum, supporting a smooth estimation of the density at x^μ, and thereby generalization, as described in equation 2.

Finally, observe that the boundary between memorization and generalization also depends on R through $\nu_{\text{in},R,\mathbf{x}}$. Naturally, this quantity will increase with R, meaning that the non-memorized region will grow with the level of noise.

4 Experimental results

4.1 Local sparsity governs memorization of samples

We now empirically investigate how local data density influences memorization behavior in diffusion models.

We train diffusion models with a U-net architecture on randomly chosen subsets of CIFAR-10 and CelebA. We then generate samples from these models and assess their closeness to training examples using different measures outlined below. Details on the experimental procedure are given in section D. We first define how we measure sparsity and memorization and then describe the outcome of the experiments shown in fig. 1 and fig. 4.

For each training sample x^μ, we quantify local sparsity using nearest-neighbor distances, which serve as a proxy for inverse local coverage. Concretely, we define

$$s(x^\mu) = \frac{1}{Nk}\sum_{i=1}^{k} \|x^\mu - x^{\mu_i}\|^2, \tag{16}$$

where $\{x^{\mu_i}\}$ are the k nearest neighbors of x^μ. Larger values of $s(x^\mu)$ correspond to sparser regions of the data distribution. We also validate that the results remain consistent for different choices of k in $\{5, 10, 20, 50\}$. The results reported in the figures correspond to $k = 10$.

Given a generated sample $\hat{x}$, we compute its distances $d_1(\hat{x})$ and $d_2(\hat{x})$ to the nearest and second-nearest training samples, and define the dominance

$$\Delta(\hat{x}) = \log \frac{d_2(\hat{x})}{d_1(\hat{x})}. \tag{17}$$

Large values of Δ indicate that a single training point dominates the local score estimation around the generated sample, while $\Delta \approx 0$ corresponds to interpolation between multiple neighbors. We assign each generated sample to its nearest training point and compute, for each x^μ, the average dominance over all generated samples assigned to it. To disentangle

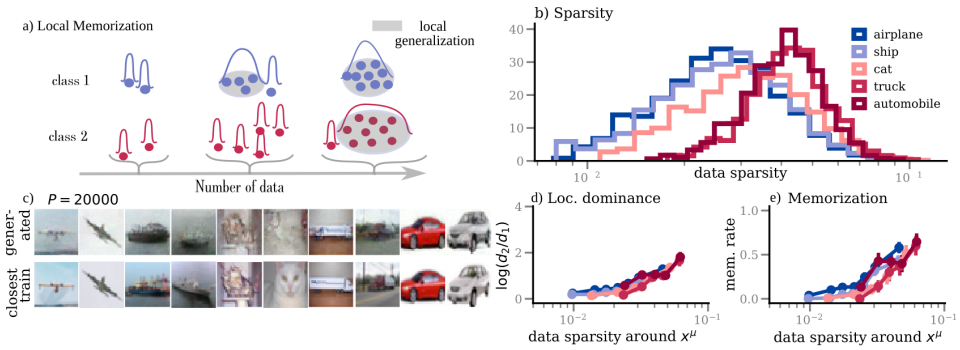


Figure 2: Intra-class sparsity predicts class-wise memorization. a) Schematic illustration of KDE for two classes with different local sparsities. b) Distribution of per-class sparsity around training points for different subclasses of CIFAR-10, measured from 20,000 training samples. c) Generated samples together with their closest training example for a diffusion model trained on $P = 20,000$ samples, sorted to contain 2 nearest neighbors for each of the four classes shown in b). d,e) Local dominance and memorization rate for $\tau = 1/3$ as a function of sparsity, binned and averaged by local sparsity. Different colors correspond to different classes.

model-specific effects from dataset geometry, we construct a baseline by replacing generated samples with held-out test data, processed in the same way. We further report a binary memorization metric used in several previous studies (Wu et al., 2025; Yoon et al., 2023; Bonnaire et al., 2026) based on the condition $d_1/d_2 < \tau$.

The middle column of fig. 1 shows the average dominance as a function of local sparsity for different training set sizes. We observe a clear monotonic increase of dominance with sparsity for generated samples: sparse regions exhibit strong single-sample dominance, while dense regions show low dominance and interpolation. Since both local dominance and sparsity depend on nearest-neighbor distances, these two quantities may be correlated. We therefore compare the local dominance measured for generated data with local dominance on a held-out test set. We observe a slight positive trend of the local dominance with local sparsity, but this trend is far weaker than that of the generated data, meaning that the increased local dominance we measure w.r.t. generated data is indeed indicative of memorization and is not explained by nearest-neighbor geometry alone.

As the number of training samples, P, increases, the overall level of dominance decreases and the dependence on sparsity weakens, consistent with the expectation that higher sample density reduces isolated regions. The corresponding results for the memorization rate are shown in the right column of fig. 1. While we do not directly estimate the threshold $\nu_{\text{in},R}(x) = -\alpha$, the observed dependence on local sparsity is consistent with a transition controlled by local coverage. For the canonical value $\tau = 0.33$ commonly used in the literature, we see a clear coexistence of memorization and generalization that depends on data sparsity. Moreover, while the precise behavior depends on τ, the same qualitative trend is observed: memorization increases with sparsity. In the appendix section C we show the outcome of the same experiment on downsampled CelebA, an image dataset of celebrity portraits. The results of these experiments are also consistent with the hypothesis. These results support a local version of the KDE picture: diffusion models exhibit a continuous transition from interpolation in dense regions to single-sample dominance in sparse regions.

4.2 Intra-class sparsity predicts class-wise memorization

We now study memorization at a coarser level, focusing on class-dependent effects. Consider a mixture of class-conditional densities ρ_c with corresponding local coverage measures $\nu_{\text{in},c}$ such that

$$\rho(x) = \textstyle\sum_c w_c \rho_c(x), \qquad \nu_{\text{in},R,c}(x) := N^{-1} \ln p_{\text{in},R,c}(x) = N^{-1} \ln \textstyle\int_{B_R(x)} d\rho_c(x'),$$

where the weights w_c sum to one, and we assume that all weights are order one in N. We now assume that $\nu_{\text{in},R,c}(x)$ remains $\mathcal{O}(1)$ in the large-N limit. Then, the local coverage is dominated by the locally densest class, i.e. $\nu_{\text{in},R}(x_N) \longrightarrow \max_c \nu_{\text{in},R,c,\mathbf{x}}$. Consequently, we obtain the same scaling behavior as in lemma 2, in a class-dependent manner:

$$z_N(x_N) \xrightarrow[N\to\infty]{\mathbb{P}} \begin{cases} \nu_{\text{in},\mathbf{x}} - v_R & \text{if } \max_c \nu_{\text{in},R,c,\mathbf{x}} > -\alpha, \\ -\infty & \text{if } \max_c \nu_{\text{in},R,c,\mathbf{x}} < -\alpha. \end{cases} \tag{18}$$

showing that memorization is governed by the class with the highest local coverage around x. On the level of individual samples, the prediction remains the same: samples in low density regions are more prone to memorization. At the class-level, this result shows that classes with higher intra-class variability (and thus lower local coverage) should be more prone to memorization. A schematic illustration is shown in fig. 2a.

We now demonstrate empirically that such class-dependent density differences naturally arise. As a concrete example, we isolate the memorization behavior of different sub-classes of CIFAR-10. In fig. 2b we show that the classes "airplane" and "ship" in CIFAR-10 have, on average, considerably lower intra-class diversity than the classes "truck" and "automobile", likely due to more homogeneous blue background colors in the former two classes. Correspondingly, we observe both higher local dominance, and higher memorization in more diverse classes. In appendix C, we show that the same trends emerge for two additional experiments.

4.3 Effective kernel width in diffusion models, training dynamics and model capacity

In the context of diffusion models, the effective kernel width at the end of the sampling trajectory should converge to zero, $R \to 0$. This would imply that all samples are memorized data, as $\lim_{R\downarrow 0} \nu_{\text{in},R,\mathbf{x}} \to -\infty$. However, in our experiments, we observed coexistence of memorized and novel samples whose memorization vulnerability is consistent with sampling from a KDE with a finite kernel width.

To understand this, we now investigate two confounding factors: first, while the KDE estimator minimizes the training loss, the diffusion model might not have been trained long enough to converge to the KDE estimator everywhere. Second, practical diffusion models might not have sufficient capacity to express the KDE solution. We measure memorization rates at different training stages. Intriguingly, we observe that during training, samples in high sparsity regions are memorized first and samples from low sparsity regions are memorized later, see fig. 3b): at roughly $2.2 \cdot 10^5$ training steps, the model initially begins to memorize isolated training data. After training roughly five times longer, at 10^6 training steps, the memorization rates have increased overall, but are still highest in regions of high sparsity. To make sense of this phenomenon, we also measure the the alignment between the output of the trained diffusion models and the optimal KDE estimator (eq. (5)), which we compute explicitly. At chance level, the alignment between two vectors in N dimensions vanishes with $N^{-\frac{1}{2}}$, therefore any finite measured cosine similarity indicates significant alignment. In general, the similarity between the KDE estimator and the trained U-nets increases as a function of training times and saturates to one (see fig. 3 c)). Differences remain only in very low noise regimes, when the similarity is evaluated over test time: at very low noise levels the diffusion models also approach the KDE solution more slowly. We provide detailed experiments across noise levels, datasets, and dataset sizes and different U-Net capacities showing consistent trends in section C.4.

These observations imply that, up to a small but finite noise level, practical diffusion models are well described by KDE, befitting the close match of the memorization profile with the KDE picture with a finite kernel width. The earlier onset of memorization of more isolated training data could then be explained by an effective kernel width that decreases as training progresses.

5 Discussion

Summary. In this work, we developed a local theory of memorisation for KDE. We showed that, in the high-dimensional limit, each point x in sample space undergoes a local transition

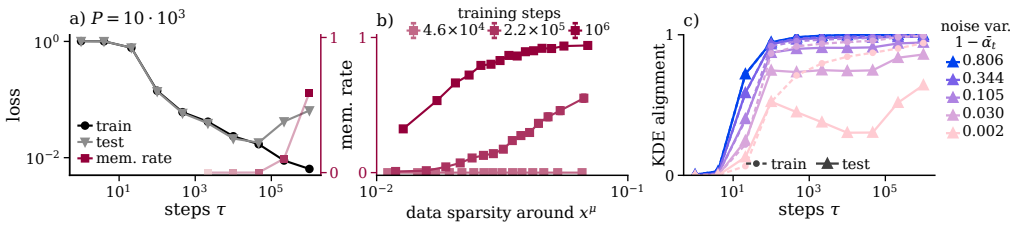


Figure 3: Training dynamics. a) Test loss, training loss, and memorized fraction of $40,000$ generated images (evaluated for $\tau = 1/3$) over training steps, for a diffusion model trained on $P = 10^4$ CIFAR-10 images. b) Memorization rate at different training stages, binned and averaged by local sparsity. c) Cosine similarity between neural network output and KDE estimator at different noise levels over training steps, evaluated at 4000 noised training/test data per point.

from memorisation to generalisation as a function of sample complexity. This transition is governed by local coverage of training data, which separates regions where the density is smoothly estimated from regions dominated by isolated training samples. This leads to a coexistence of generalization and memorization within the same model. Our analysis further predicts that, in multi-class settings, the behavior at a point x is controlled by the class with the highest local coverage. Empirically, we find consistent evidence for these predictions in diffusion models: memorization increases with local sparsity and intra-class variability, and different classes exhibit systematically different memorization behavior. Taken together, these results show that memorization is not a global failure mode, but a localized phenomenon driven by the geometry of the data distribution. In particular, isolated regions of the data space can be memorized even when the majority of generated samples are novel.

Limitations. On the theoretical side, our analysis predicts an abrupt transition between memorization and generalization as a function of local coverage and kernel width in the infinite dimensional limit, whereas empirically we observe a more gradual dependence on local sparsity. This more gradual behavior is likely the result of finite size fluctuations ($N, P < \infty$) and the fact that, in practical settings, architectural constraints (Yoon et al., 2023; Kamb & Ganguli, 2025; George et al., 2025a; Zhang et al., 2025) and early stopping the training process (Favero et al., 2025; Bonnaire et al., 2026) incur differences between learned solutions and KDE, as our experiments in section 4.3 and section C.4 also demonstrate.

These experiments yield an interesting observation: local coverage also predicts which data are memorized first as a function of training time. On the high level, this result appears to be in opposition to Rodriguez et al. (2026)'s recent work asserting that "typical" data are memorized first. However, their analysis considered only high-level features of data such as class membership or attributes, whereas our experiments in section 4.2 show that the sparsity within a class, or a collection of data with shared attributes, is also highly relevant to memorization. A natural next question is therefore predicting how memorization depends on both local sparsity and training time.

One direction for further research lies in determining the potential lower-dimensional metric spaces that diffusion models may implicitly or explicitly Rombach et al. (2022) operate in. Such a dimensionality reduction may also explain the reduced (non-exponential) sample complexity observed in practice: as Achilli et al. (2025b) already show on the global level, the required sample complexity should scale with this reduced dimension. In such cases, the effective kernel also should be understood as acting in this learned metric space.

Finally, while the overall agreement between KDE estimation and trained diffusion models appears to be strong, differences remain at very low noise levels and for reduced model sizes. While we found consistently that diffusion models that memorize are more similar to the KDE solution than those that do not memorize, and thereby the memorization-vulnerability predictions from KDE translate consistently, this does not mean that diffusion model generalization can be fully explained within the same theoretical picture. We believe that these differences are worth studying in the future.

References

Beatrice Achilli, Luca Ambrogioni, Carlo Lucibello, Marc Mézard, and Enrico Ventura. The Capacity of Modern Hopfield Networks under the Data Manifold Hypothesis, March 2025a. URL http://arxiv.org/abs/2503.09518. arXiv:2503.09518 [cond-mat].

Beatrice Achilli, Luca Ambrogioni, Carlo Lucibello, Marc Mézard, and Enrico Ventura. Memorization and generalization in generative diffusion under the manifold hypothesis. Journal of Statistical Mechanics: Theory and Experiment, 2025(7):073401, July 2025b. ISSN 1742-5468. doi: 10.1088/1742-5468/ade136. URL https://doi.org/10.1088/1742-5468/ade136.

Beatrice Achilli, Enrico Ventura, Gianluigi Silvestri, Bao Pham, Gabriel Raya, Dmitry Krotov, Carlo Lucibello, and Luca Ambrogioni. Losing dimensions: Geometric memorization in generative diffusion, March 2026. URL http://arxiv.org/abs/2410.08727. arXiv:2410.08727 [stat].

Luca Ambrogioni. In Search of Dispersed Memories: Generative Diffusion Models Are Associative Memory Networks. Entropy, 26(5):381, 2024. ISSN 1099-4300. doi: 10.3390/e26050381. URL https://www.mdpi.com/1099-4300/26/5/381.

Giulio Biroli and Marc Mézard. Kernel Density Estimators in Large Dimensions, October 2024. URL http://arxiv.org/abs/2408.05807. arXiv:2408.05807 [cs].

Giulio Biroli, Tony Bonnaire, Valentin de Bortoli, and Marc Mézard. Dynamical regimes of diffusion models. Nature Communications, 15(1):9957, November 2024. ISSN 2041-1723. doi: 10.1038/s41467-024-54281-3. URL https://www.nature.com/articles/s41467-024-54281-3.

Tony Bonnaire, Raphaël Urfin, Giulio Biroli, and Marc Mézard. Why diffusion models don't memorize: The role of implicit dynamical regularization in training. Advances in Neural Information Processing Systems, 38:141266–141286, 2026.

Nicholas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramèr, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting Training Data from Diffusion Models, 2023. URL http://arxiv.org/abs/2301.13188. arXiv:2301.13188.

Li Deng. The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web]. IEEE Signal Processing Magazine, 29(6):141–142, November 2012. ISSN 1558-0792. doi: 10.1109/MSP.2012.2211477. URL https://ieeexplore.ieee.org/document/6296535.

V. Dotsenko. Replica solution of the random energy model. EPL (Europhysics Letters), 95(5):50006, September 2011. ISSN 0295-5075, 1286-4854. doi: 10.1209/0295-5075/95/50006. URL https://iopscience.iop.org/article/10.1209/0295-5075/95/50006.

Zhengyu Fang, Zhimeng Jiang, Huiyuan Chen, Xiaoge Zhang, Kaiyu Tang, Xiao Li, and Jing Li. A Closer Look on Memorization in Tabular Diffusion Model: A Data-Centric Perspective, August 2025. URL http://arxiv.org/abs/2505.22322. arXiv:2505.22322 [cs].

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

Jerome Garnier-Brun, Luca Biggio, Davide Beltrame, Marc Mézard, and Luca Saglietti. Biased Generalization in Diffusion Models, March 2026. URL http://arxiv.org/abs/2603.03469. arXiv:2603.03469 [cs].

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].

Anand Jerry George, Rodrigo Veiga, and Nicolas Macris. Denoising score matching with random features: Insights on diffusion models from precise learning curves. In The 29th International Conference on Artificial Intelligence and Statistics, 2026. URL https://openreview.net/forum?id=ZnplHm2uRt.

Xiangming Gu, Chao Du, Tianyu Pang, Chongxuan Li, Min Lin, and Ye Wang. On Memorization in Diffusion Models, February 2025. URL http://arxiv.org/abs/2310.02664. arXiv:2310.02664 [cs].

Dongjae Jeon, Dueun Kim, and Albert No. Understanding and Mitigating Memorization in Generative Models via Sharpness of Probability Landscapes, August 2025. URL http://arxiv.org/abs/2412.04140. arXiv:2412.04140 [cs].

Zahra Kadkhodaie, Florentin Guth, Eero Simoncelli, and Stéphane Mallat. Generalization in diffusion models arises from geometry-adaptive harmonic representations. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), International Conference on Learning Representations, volume 2024, pp. 46543–46567, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/cbaf319a4712385b5ba8a414808b5713-Paper-Conference.pdf.

Mason Kamb and Surya Ganguli. An analytic theory of creativity in convolutional diffusion models. June 2025. URL https://openreview.net/forum?id=ilpL2qACla.

Juyeop Kim, Songkuk Kim, and Jong-Seok Lee. How Diffusion Models Memorize, September 2025. URL http://arxiv.org/abs/2509.25705. arXiv:2509.25705 [cs].

A. Krizhevsky. Learning Multiple Layers of Features from Tiny Images. 2009. URL https://www.semanticscholar.org/paper/Learning-Multiple-Layers-of-Features-from-Tiny-Krizhevsky/5d90f06bb70a0a3dced62413346235c02b1aa086.

Sixu Li, Shi Chen, and Qin Li. A Good Score Does not Lead to A Good Generative Model, January 2024. URL http://arxiv.org/abs/2401.04856. arXiv:2401.04856 [cs].

Carlo Lucibello and Marc Mézard. Exponential Capacity of Dense Associative Memories. Physical Review Letters, 132(7):077301, February 2024. doi: 10.1103/PhysRevLett.132.077301. URL https://link.aps.org/doi/10.1103/PhysRevLett.132.077301.

Antoine Maillard and Sebastian Goldt. Memorisation, convergence and generalisation in generative models. arXiv preprint arXiv:2605.21402, 2026.

Bao Pham, Gabriel Raya, Matteo Negri, Mohammed J. Zaki, Luca Ambrogioni, and Dmitry Krotov. Memorization to Generalization: The Emergence of Diffusion Models from Associative Memory. November 2024. URL https://openreview.net/forum?id=zVMMaVy2BY.

Jakiw Pidstrigach. Score-Based Generative Models Detect Manifolds. October 2022. URL https://openreview.net/forum?id=AiNrnIrDfD9.

Marta Aparicio Rodriguez, Anastasia Borovykh, Grigorios A. Pavliotis, and Daniel J. Korchinski. Diffusion Models Preferentially Memorize Prototypical Examples or: Why Does My Diffusion Model Love Slop?, May 2026. URL http://arxiv.org/abs/2605.30642. arXiv:2605.30642 [cs.LG].

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-Resolution Image Synthesis with Latent Diffusion Models, April 2022. URL http://arxiv.org/abs/2112.10752. arXiv:2112.10752 [cs.CV].

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].

Brendan Leigh Ross, Hamidreza Kamkari, Tongzi Wu, Rasa Hosseinzadeh, Zhaoyan Liu, George Stein, Jesse C. Cresswell, and Gabriel Loaiza-Ganem. A Geometric Framework for Understanding Memorization in Generative Models, March 2025. URL http://arxiv.org/abs/2411.00113. arXiv:2411.00113 [stat].

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].

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Understanding and Mitigating Copying in Diffusion Models. November 2023. URL https://openreview.net/forum?id=HtMXRGbUMt.

Binxu Wang, Jacob Zavatone-Veth, and Cengiz Pehlevan. A random matrix theory perspective on the consistency of diffusion models. arXiv preprint arXiv:2602.02908, 2026.

Yuxin Wen, Yuchen Liu, Chen Chen, and Lingjuan Lyu. DETECTING, EXPLAINING, AND MITIGATING MEMO- RIZATION IN DIFFUSION MODELS. 2024.

Yu-Han Wu, Pierre Marion, Gérard Biau, and Claire Boyer. Taking a Big Step: Large Learning Rates in Denoising Score Matching Prevent Memorization. In Proceedings of Thirty Eighth Conference on Learning Theory, pp. 5718–5756. PMLR, July 2025. URL https://proceedings.mlr.press/v291/wu25a.html.

TaeHo Yoon, Joo Young Choi, Sehyun Kwon, and Ernest K. Ryu. Diffusion Probabilistic Models Generalize when They Fail to Memorize. July 2023. URL https://openreview.net/forum?id=shciCbSk9h#all.

Zekai Zhang, Xiao Li, Xiang Li, Lianghe Shi, Meng Wu, Molei Tao, and Qing Qu. Generalization of Diffusion Models Arises with a Balanced Representation Space. October 2025. URL https://openreview.net/forum?id=57THeGgNAN.

Appendix

A Further related Work

Impact of training setup on memorization A growing body of empirical and theoretical work has established that memorization depends strongly on properties of the training setup. In particular, it is exacerbated by data duplication (Carlini et al., 2023), prompt conditioning and classifier-free guidance (Somepalli et al., 2023; Wen et al., 2024; Kim et al., 2025; Gu et al., 2025), model capacity (Yoon et al., 2023; George et al., 2025b; Zhang et al., 2025), dataset structure (Somepalli et al., 2023; Kim et al., 2025; Gu et al., 2025; Yoon et al., 2023), and training duration (Bonnaire et al., 2026; Favero et al., 2025; Garnier-Brun et al., 2026), while decreasing with learning rate (Wu et al., 2025) and dataset size (Somepalli et al., 2023; Kadkhodaie et al., 2024; Maillard & Goldt, 2026).

Local density estimators and memorization in practice Strategies to measure and mitigate memorization include determining the local intrinsic dimension of generated points (Ross et al., 2025), and estimating how "peaked" the estimated density is around a given point (Jeon et al., 2025) as well as identifying memorized prompts (Wen et al., 2024; Kim et al., 2025). These strategies share a common intuition: isolated data points produce a highly peaked local density, and produce samples of low intrinsic dimension. This perspective is naturally consistent with a kernel density estimation (KDE) view of diffusion models. Isolated training points induce highly concentrated local density estimates and generate low-dimensional samples, while dense regions support smoother interpolation. Similarly, conditioning can be interpreted as restricting the effective sample set contributing to the

density estimate, thereby increasing local sparsity and promoting memorization. In this sense, existing empirical methods already implicitly rely on a local density perspective on memorization.

Memorization in kernel density estimation On the theoretical side, several works have established a connection between diffusion models and kernel density estimation (Pidstrigach, 2022; Pham et al., 2024; Ambrogioni, 2024; Biroli & Mézard, 2024; Achilli et al., 2025a; Li et al., 2024; Lucibello & Mézard, 2024). Within this framework, one can determine a global "collapse phase" in diffusion models as a function of effective noise in the backward/sampling process (Biroli et al., 2024), where the model begins to reproduce training data. More generally, the sample complexity required for smooth interpolation with kernel density estimators is known to scale exponentially with the ambient or manifold dimension(Lucibello & Mézard, 2024; Achilli et al., 2025b) of the data. Recent work has further shown that memorization can emerge gradually through a loss of manifold dimensions Achilli et al. (2026) and that this process may be spatially inhomogeneous. Existing theoretical analyses on KDE remain largely global: they characterize when a model as a whole memorizes or generalizes, but do not predict which specific regions or samples are more likely to be memorized. In contrast, we develop a local theory of memorization that explicitly links memorization to data coverage at the level of individual points. This theory provides a principled explanation for the heterogeneous memorization patterns observed in practice.

B Details on Theoretical results

In this appendix, we give a detailed proofs of the results presented in section 3.

B.1 Convergence of the exponential transform

To establish the behavior of the KDE, we characterize the cumulative distribution function (CDF) of the log-density z through the exponential transform eq. (8). At the base of theorem 1 therefore lies the convergence of the exponential transform to the CDF, which is provided by the following Lemma.

Lemma 3 (Convergence of the exponential transform to the CDF.). Let $N \in \mathbb{N}, N > 0, x_N \in R^N$ and $F_N(y) = \Pr(z_N(x_N) \leq y)$ be the cumulative distribution function of $z_N(x_N)$. Then for $\overline{W_N(y, x_N)}$ defined as in eq. (8) and $\delta > 0$, we find that

$$e^{-e^{-N\delta}} F_N(y-\delta) \leq \overline{W_N(y, x_N)} \leq F_N(y+\delta) + e^{-e^{N\delta}} \Pr(z_N \geq y+\delta).. \tag{19}$$

Consequently, if the limit $F_N \to F_{\mathbf{x}}$ exists and if y is a continuity point of $F_{\mathbf{x}}$, then

$$\overline{W_N(y, x_N)} \to F_{\mathbf{x}}(y).$$

Proof. For notational simplicity we suppress the dependence on x and write

$$z_N := z_N(x), \qquad W_N(y) := \exp\{-\exp[-N(y - z_N)]\}.$$

We first show the lower bound. If $z_N \leq y - \delta$, then

$$W_N(y) = \exp\{-\exp[-N(y - z_N)]\} \geq \exp\{-e^{-N\delta}\}.$$

In other words, using the indicator function $\mathbf{1}$, we know that

$$W_N(y) \geq e^{-e^{-N\delta}} \mathbf{1}\{z_N \leq y - \delta\}.$$

Taking expectations gives

$$\overline{W_N(y)} \geq e^{-e^{-N\delta}} \mathbb{P}(z_N \leq y - \delta).$$

We now take care of the upper bound. Since $0 \leq W_N(y) \leq 1$, we know that

$$\begin{aligned} W_N(y) &\leq \mathbf{1}\{z_N < y + \delta\} + W_N(y)\mathbf{1}\{z_N \geq y + \delta\} \\ &\leq \mathbf{1}\{z_N < y + \delta\} + e^{-e^{N\delta}} \mathbf{1}\{z_N \geq y + \delta\}. \end{aligned}$$

Taking the average yields

$$\overline{W_N(y)} \leq \mathbb{P}(z_N < y + \delta) + e^{-e^{N\delta}} \mathbb{P}(z_N \geq y + \delta).$$

Combining the two bounds yields Lemma 3. □

B.2 Average over draws of samples theorem 1

We now proceed with the proof of theorem 1. Let us first keep $N \in \mathbb{N}, N > 0, x_N \in R^N$ fixed. Our starting point is eq. (8). Recall that the notation $\overline{f(z)}$ denotes the average over the data set, meaning that when we insert the definition of $z(x)$ and explicitly state the average, we find

$$\begin{aligned}\overline{W_N(y, x_N)} &= \int \prod_\mu d\rho(x^\mu) \exp\left(-e^{-N(y+\alpha)} \sum_\mu K(x - x^\mu)\right) \\ &= \left\{\int d\rho(\tilde{x}) \exp\left[-e^{-N(y+\alpha)} K\left(x - \tilde{x}\right)\right]\right\}^P\end{aligned}$$

which directly yields eq. (9). Taking the limit then reveals the three scaling regimes

$$\overline{W_N(y, x_N)} \to \begin{cases} 0 & \text{if } i_{\mathbf{x}}(y) = \infty, \\ \exp[-i_{\mathbf{x}}(y)] & \text{if } 0 < i_{\mathbf{x}}(y) < \infty, \\ 1 & \text{if } i_{\mathbf{x}}(y) = 0. \end{cases} \tag{20}$$

Using, lemma 3, we then find that the limit of the exponential transform yields the limiting CDF. □

B.3 Hard spherical kernels

We now turn to the proof of Lemma 2. Using the definition of p_{in}, we find that the integral I_N defined in eq. (9) decomposes into two areas: those where $K = 0$ and those where $K = 1/V_R$. Using this distinction, we then find

$$I_N\left(y, x_N\right) = p_{\text{in},N}(x_N)\left(1 - \exp\left(-\exp\left\{-N\left(y + \alpha + \frac{1}{N}\ln V_R\right)\right\}\right)\right) \tag{21}$$

Let us now assume that $\frac{1}{N}\ln V_R \to v_R$ converges, we can then absorb this constant into $y_R = y + v_R$. To apply theorem 1, we use the assumption that $\nu_{\text{in},R,\mathbf{x}}$ exists and compute the limit eq. (10) in two distinct regimes: $y_R + \alpha > 0$ and $y_R + \alpha < 0$.

In the first regime, $y_R + \alpha = \delta > 0$, we find the leading order behavior $e^{-e^{-N\delta}} = 1 - e^{-N\delta} + \mathcal{O}(e^{-2N\delta})$ and therefore

$$i_{\mathbf{x}} = \lim_{N\to\infty} e^{N(\alpha + \nu_{\text{in},N}(x_N))}\left(1 - e^{-e^{-N\delta}}\right) = \begin{cases} 0 & \text{if } y_R - \nu_{\text{in},R,\mathbf{x}} > 0 \\ \infty & \text{if } y_R - \nu_{\text{in},R,\mathbf{x}} < 0 \end{cases}.$$

In the second regime, $y_R + \alpha = -\delta < 0$, we find

$$i_{\mathbf{x}} = \lim_{N\to\infty} e^{N(\alpha + \nu_{\text{in},N}(x_N))}\left(1 - e^{-e^{N\delta}}\right) = \begin{cases} 0 & \text{if } \alpha + \nu_{\text{in},R,\mathbf{x}} < 0 \\ \infty & \text{if } \alpha + \nu_{\text{in},R,\mathbf{x}} > 0 \end{cases}.$$

Applying theorem 1 to both regimes, we find

$$F_{\mathbf{x}}(y) = \begin{cases} \Theta\left(y_R - \nu_{\text{in},R,\mathbf{x}}\right) & y_R + \alpha > 0 \\ \Theta\left(-\left(\nu_{\text{in},R,\mathbf{x}} + \alpha\right)\right) & y_R + \alpha < 0 \end{cases}$$

which yields the probability that $z \leq y$ in the limit, thanks to lemma 3. The first line shows that when y_R is larger than $-\alpha$, the CDF becomes a step function with a jump at $y = \nu_{\text{in},R,\mathbf{x}}$.

We split the second line into two scenarios, $\nu_{\text{in},R,\mathbf{x}} + \alpha > 0$ and $\nu_{\text{in},R,\mathbf{x}} + \alpha < 0$ In the first, scenario $\nu_{\text{in},R,\mathbf{x}} > -\alpha > y$, hence this implies $y_R < \nu_{\text{in},R}(x)$, and the limiting CDF is zero in y_R. On then other hand, in the second scenario, when $\nu_{\text{in},R,\mathbf{x}} < -\alpha$, then the CDF is one for any finite value of y_R. This means that z is smaller than any finite value of y_R with probability one. Taken together, the first and the second line imply that the CDF has a

jump at $y_R = \nu_{\text{in},R,\mathbf{x}}$ when $\nu_{\text{in},R,\mathbf{x}} > -\alpha$. The second scenario in the second line implies that when $\nu_{\text{in},R,\mathbf{x}} < -\alpha$, all draws of z are smaller than any finite y_R, hence $z \to -\infty$.

Consequently, we find that the cumulative distribution either has the shape of a step function with a jump in $y_R = \nu_{\text{in},R} \Rightarrow y = \nu_{\text{in},R} \Rightarrow -v_R$ when $\nu_{\text{in},R}(x) > -\alpha$. Hence the value of $z_N(x_N)$ concentrates on the "jump location". Or the cumulative distribution function is equal to one at every finite value, meaning that all intervals in which z_N can occur with finite probability are infinitely small. Hence

$$\begin{cases} \nu_{\text{in},R}(x) - v_R & \text{if } \nu_{\text{in},R}(x) > -\alpha \\ -\infty & \text{if } \nu_{\text{in},R}(x) < -\alpha \end{cases} \tag{22}$$

which finally yields lemma 2. □

B.4 Relation to Gumbel-type extreme value statistics

In this appendix we clarify the relation between the exponential transform used in Eq. (10) and Gumbel-type extreme value statistics. Let us first clarify what we mean by Gumbel-type extreme value statistics: for independent random variables $u_1, \ldots, u_P$, one has

$$\mathbb{P}\left(\max_\mu u_\mu < y\right) = \left[1 - \mathbb{P}(u_\mu \geq y)\right]^P . \tag{23}$$

which is the standard structure associated with Gumbel-type extreme value statistics and, identifying I_N with $\mathbb{P}(u_\mu \geq y)$, has strong similarity with eq. (9).

We will first show that letting $N \to \infty$ at $P < \infty$ fixed, W_N has a natural extreme-value interpretation. However, this does not mean that the calculation replaces the KDE by its largest term, because in the main result, we let $P \to \infty$ with $\frac{1}{N} \ln P = \alpha$ fixed. In this case, W_N is a soft probe of the cumulative distribution function of the full log-KDE.

For the sake of comparison, however, let us now keep $P < \infty$ and all samples fixed. We then find that W_N can be rewritten as

$$W_N(y, x) = \prod_{\mu=1}^{P} \exp\left\{-K(x - x^\mu) e^{-N(y+\alpha)}\right\}. \tag{24}$$

Now define the single-sample contribution $u_\mu(x) := \frac{1}{N} \ln K(x - x^\mu) - \alpha$. Then each factor in Eq. equation 24 can be written as

$$\exp\left\{-K_\mu(x) e^{-N(y+\alpha)}\right\} = \exp\left\{-\exp[N(u_\mu(x) - y)]\right\}. \tag{25}$$

If we were to fix P and all $u_\mu(x)$, and let $N \to \infty$ this converges to a hard threshold:

$$\exp\left\{-\exp[N(u_\mu(x) - y)]\right\} \xrightarrow[N\to\infty]{} \Theta(y - u_\mu(x)). \tag{26}$$

Consequently, if this hard-threshold limit is taken termwise before accounting for the exponentially many samples, Eq. equation 24 becomes

$$W_N(y, x) \approx \prod_{\mu=1}^{P} \Theta(y - u_\mu(x)) = \Theta\left(y - \max_\mu u_\mu(x)\right). \tag{27}$$

This is the origin of the formal similarity with Gumbel's law: the product form resembles the cumulative distribution function of a maximum. In regimes where the KDE is dominated by a single large kernel contribution, $z_N(x)$ is well approximated by

$$\widetilde{z}_N(x) := \max_\mu \frac{1}{N} \log \frac{K_\mu(x)}{P} = \max_\mu u_\mu(x). \tag{28}$$

In such max-dominated regimes, the above Gumbel-type interpretation directly describes the statistics of $z_N(x)$.

However, to obtain a generalization regime, we must send P to infinity too. Consequently, in the generalization regime, exponentially many samples may contribute to the KDE at

any point. Then the sum cannot be replaced by its largest term. This distinction can be seen explicitly for the hard-sphere kernel. Let $m_N(x)$ be the number of training samples inside the corresponding ball around x with volume $V_R = 1$. Then

$$Z_N(x) = \frac{m_N(x)}{P}, \qquad \Rightarrow \qquad z_N(x) = \frac{1}{N} \log m_N(x) - \alpha. \tag{29}$$

If the local mass of the ball satisfies $\ln p_{\text{in}}(x) = N\nu_{\text{in}}(x) + o(N)$, then the typical number of samples in the ball is

$$m_N(x) \sim P p_{\text{in}}(x) = e^{N(\alpha+\nu_{\text{in}}(x))+o(N)} \tag{30}$$

whenever $\alpha + \nu_{\text{in}}(x) > 0$. In this regime,

$$z_N(x) \to \nu_{\text{in}}(x), \tag{31}$$

up to subexponential prefactors. By contrast, the largest single contribution is only

$$\widetilde{z}_N(x) = -\alpha \tag{32}$$

provided that at least one sample lies inside the ball. Thus, whenever $\nu_{\text{in}}(x) > -\alpha$, the full KDE exponent $z_N(x)$ differs from the maximum contribution $\widetilde{z}_N(x)$.

The Gumbel-like structure of Eq. (11) should therefore be understood as follows: the exponential transform is a soft thresholding device which, after factorization over samples, resembles an extreme-value observable. If one hardens this threshold before accounting for the exponentially many samples, one obtains a maximum-probe interpretation. Keeping the transform soft through the large-N calculation, however, retains the collective contribution of exponentially many moderate terms and therefore probes the full log-KDE $z_N(x)$.

C Additional Experiments

C.1 Local sparsity and dominance

In fig. 4, we report the analogous experiment to fig. 1 for CelebA data, which we downsample to 32×32 greyscale pixels. We find that both local dominance and memorization increase with local sparsity. In comparison to CIFAR-10, we observe that the CelebA dataset appears to have fewer samples in very low density regions (compare logarithmic scale of fig. 4 d) to fig. 1d).

C.2 Attribute-dependent Memorization, CelebA

The CelebA dataset consists of celebrity portraits that are annotated with "attributes" such as hair color or accessories. We construct four classes from these attributes, conditioning on "blond" and "not blond" as well as "wearing hat" and the opposite. Again, we find that local dominance (and thus memorization) increases with data sparsity, see fig. 5. Samples from classes where "wearing hat" is true are typically more diverse. Therefore this class has a higher average sparsity per class,and is therefore more likely to be memorized. These classes are not balanced: approximately one sixth of images has attribute "blond" and approximately 5 % of images have attribute "wearing hat". Consequently the diffusion model generates fewer samples that are memorized data points with these attributes, and even fewer samples are closest to training data where both attributes are true. This is reflected in the lower bin resolution in fig. 5b) and fewer such points appearing in fig. 5 c).

C.3 Class-dependent Memorization, CIFAR-10+MNIST

In a separate experiment, we train diffusion models on a mixture of MNIST (Deng, 2012) and CIFAR-10 images. MNIST, consisting of handwritten digits, is structurally simpler and occupies a much more concentrated region of image space, whereas CIFAR-10 exhibits substantially higher variability. Treating MNIST and CIFAR-10 as two distinct classes, we observe that MNIST samples consistently exhibit lower local sparsity than CIFAR-10 samples. As predicted by the theory, this translates into reduced memorization: in fig. 6, both local dominance and memorization rates increase with sparsity. For the comparison of

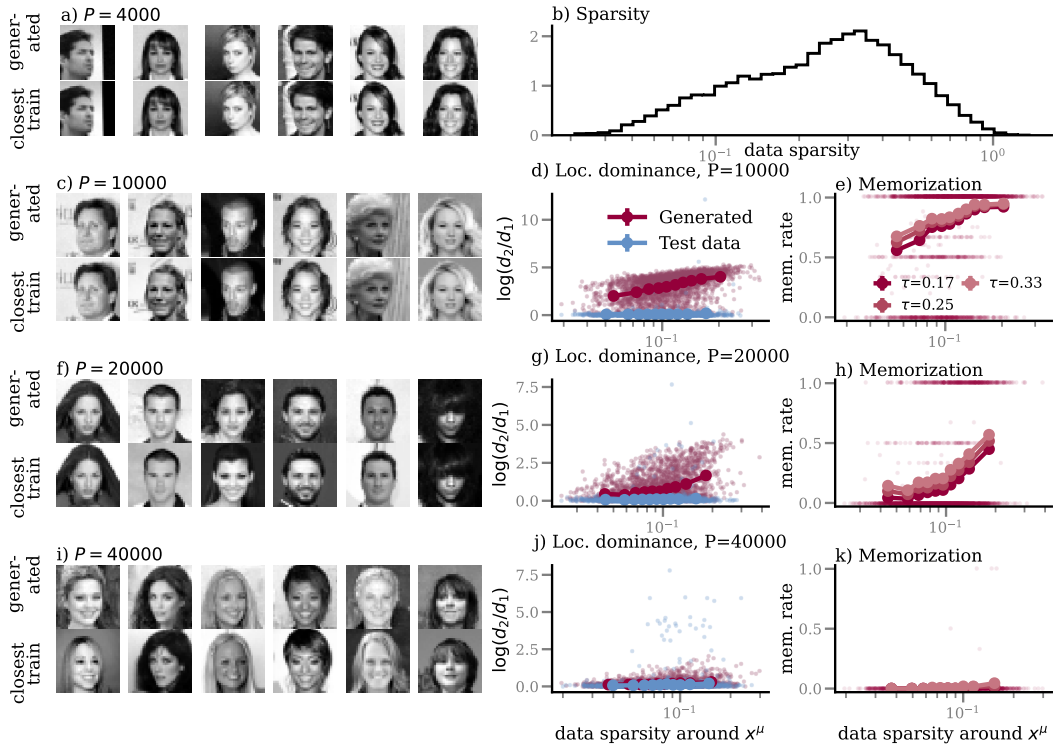


Figure 4: Local memorization in CelebA data. a) Generated samples + closest training image (measured by cosine similarity for diffusion model trained on $P = 4,000$ training examples. b) Distribution of data sparsity for 40,000 samples from CelebA. c) Generated samples + closest training image (measured by cosine similarity for diffusion model trained on $P = 10,000$ training examples. d) Local dominance as a function of sparsity. For each generated sample $\hat{x}$, we compute distances d_1 and d_2 to its nearest and second-nearest training samples, and define dominance via their ratio. Each point corresponds to a training sample x^μ, aggregating generated samples assigned to it. Scatter shows raw values, markers indicate binned averages. Blue points show the same quantity computed using test data, providing a baseline without memorization. e) Memorization rate per training sample x^μ, defined as the fraction of generated samples satisfying $d_1/d_2 < \tau$. Scatter shows raw values for $\tau = 0.1$, markers indicate binned averages across thresholds. f)–k) Same measurements for models trained on larger datasets. As the number of training samples increases, both dominance and memorization decrease, and their dependence on sparsity weakens. Overall, sparse regions exhibit strong single-sample dominance and higher memorization, while dense regions promote interpolation across multiple training points.

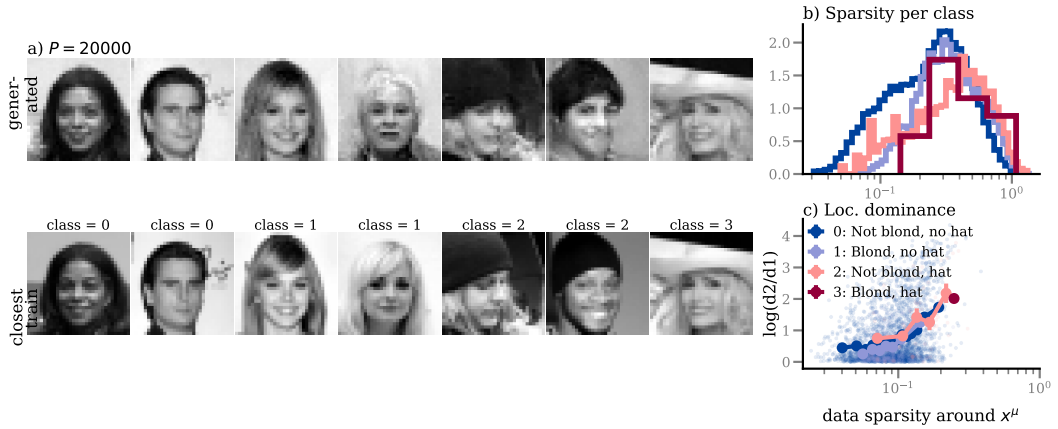


Figure 5: Attribute-conditioned memorization in CelebA. a) Generated samples + closest training image (measured by cosine similarity) for diffusion model trained on $P = 20000$ training examples, sorted by classes. b) Distribution of data sparsity for 40000 samples from CelebA. d) Local dominance of training sample x^μ against local data sparsity around x^μ for a diffusion model trained on $P = 20000$ training examples from CelebA.

CIFAR and MNIST, all distances are computed on ℓ_2-normalized samples, so that sparsity reflects relative geometric structure (angular similarity) rather than differences in overall scale. We verified that using unnormalized samples yields qualitatively similar results. On average, MNIST points lie in denser regions and are therefore memorized less than CIFAR-10 points.

C.4 Training Dynamics and Model Sizes

Here, we show additional experiments regarding memorization and the similarity to the KDE picture using different U-Net model sizes and training dynamics. We begin with showing the memorization rate for different training stages across sample complexities on CelebA data, to supplement fig. 3 b), which we show in fig. 7. We here report local dominance only, as memorization rates would be simply a thresholded average of the former. In fig. 7, we find consistently with the experiment on CIFAR-10 data, that as a function of training set size, and number of training steps, isolated samples are memorized first: we observe consistently across three dataset sizes that isolated data in high sparsity regions have higher local coverage in earlier training stages, i.e. they are memorized first. At the same time, the onset of memorization is further delayed with the number of training data.

Next, we inspect the similarity between the trained neural networks and the KDE score. For this comparison, we compute the optimal KDE score explicitly and compare its cosine similarity with the network output for noised training and test data, at different levels of noise $\sqrt{\bar{\alpha}_t}$, see fig. 8. We also vary the size of the trained networks, by removing intermediate layers from the standard U-Net, this yields three different U-Net sizes: "max.", which corresponds to the standard U-Net architecture we use throughout the paper, "med." and "min.", which are smaller versions of the same architecture. We use the same training protocol on all three architectures and report the cosine similarity to the KDE score in fig. 8.

Consistent with fig. 3 c), we find that the simialrity with the KDE solution tends to grow as a function of training steps and that the agreement is strongest at higher levels of noise. Moreover, reducing the model size leaves the alignment with the KDE score at higher levels of noise unaffected, while it decreases the alignment between the two solutions at lower levels of noise - matching the intuition that smaller networks may not have the capacity to fit the KDE solution and therefore deviate from it. Nevertheless, even in the very low noise regimes, the similarities between learned solutions of the reduced networks and KDE are far above chance level.

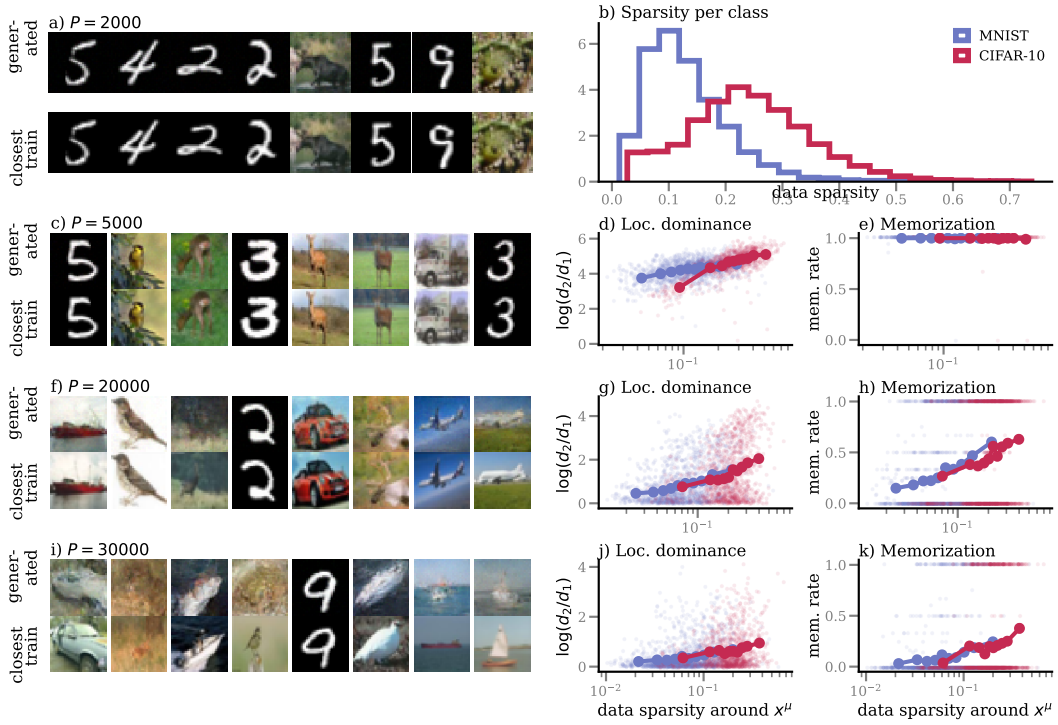


Figure 6: Class-wise memorization on combined MNIST and CIFAR-10. a) Generated samples together with their closest training example (measured by cosine similarity) for a diffusion model trained on $P = 2000$ samples. b) Distribution of per-class sparsity around training points for four classes in CIFAR-10 measured from 20,000 training samples. c) Generated samples together with their closest training example (measured by cosine similarity) for a diffusion model trained on $P = 5000$ samples. d) Class-wise local dominance of a training point x^μ as a function of local data sparsity. Local dominance is defined via the ratio between the distance d_1 to the closest generated sample and the distance d_2 between that generated sample and the second-closest training point. MNIST points are shown in blue, CIFAR-10 points in red. e) Memorization rate of each sample x^μ, defined as the fraction of generated samples satisfying $d_1/d_2 < \tau = 1/3$. Different classes are shown as different colors. f)–k) Same measurements for models trained on larger datasets. As the number of training samples increases, both dominance and memorization decrease, and their dependence on sparsity weakens.

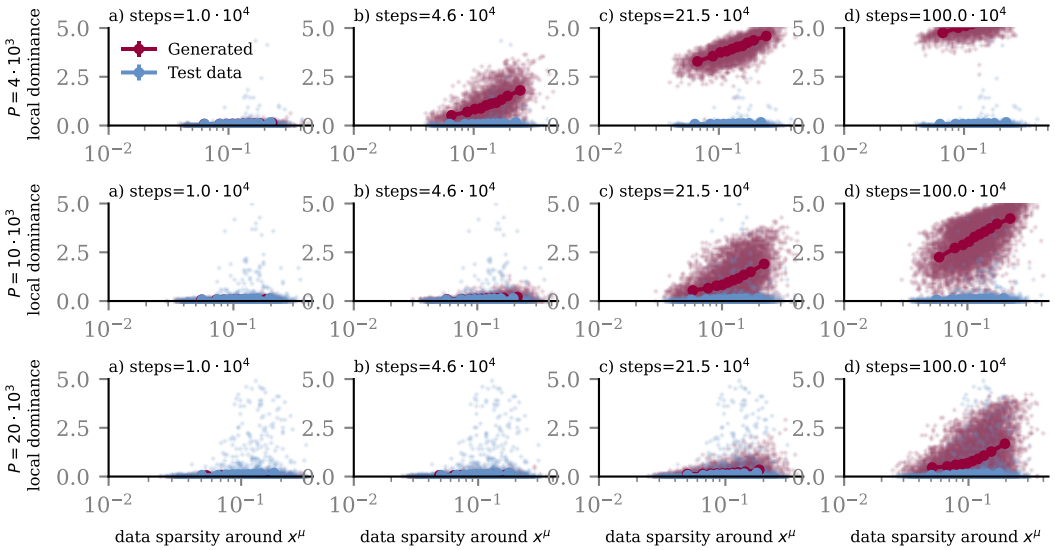


Figure 7: Local dominance as a function of local sparsity, for different training stages of the diffusion model, trained on downsampled CelebA data as in section C.1. Left to right: increasing number of training steps. Top to bottom: increasing training set size P. To compute the local dominance, we generated $40,000$ samples per training stage and P.

Finally, we report the local dominance for the three different network architectures after 10^6 training steps in fig. 9. We find that the larger the capacity, the stronger the memorization: The smallest U-Net produces samples whose local dominance is comparable to the test set baseline. The two larger models have very high local dominance at the smallest training set size, but at larger P the "medium" sized U-Net shows a local dominance profile consistent with co-occuring memorization and novel sample production while the largest U-Net is still producing only memorized data. These observations, together with the measured alignment to the KDE solution, are again consistent with the hypothesis that memorization vulnerability is described by a KDE estimation whose effective kernel width R decreases as the model capacity increases.

D Experimental Details

Model. We train denoising diffusion models using a standard discrete-time formulation with $T = 1000$ timesteps. The score network is parameterized by a U-Net architecture Ronneberger et al. (2015) with convolutional filters. All models operate directly in pixel space.

Training procedure. Models are trained using the standard diffusion objective with mean squared error loss. Optimization is performed using Adam with learning rate 10^{-4} and batch size 100. Training proceeds for a fixed number of 10^6 gradient steps for all models.

Evaluation For all experiments, the dataset is deterministically split into a training set of size P and a held-out test set. The same split is reused for all evaluations to ensure comparability across checkpoints and metrics. At evaluation time, we generated between $N_{\text{gen}} = 4000$ and $N_{\text{gen}} = 40000$ samples from the trained diffusion model. Figures fig. 1, fig. 2, fig. 3, fig. 9, fig. 7 report data obtained for $N_{\text{gen}} = 40000$. Only figures fig. 4, fig. 6 report data obtained for the reduced sampled size of $N_{\text{gen}} = 4000$, the latter reduction was undertake to reduce the computational cost of the experiments, given the consistency of the trends observed across the figures, we do not expect it to significantly alter the results of the

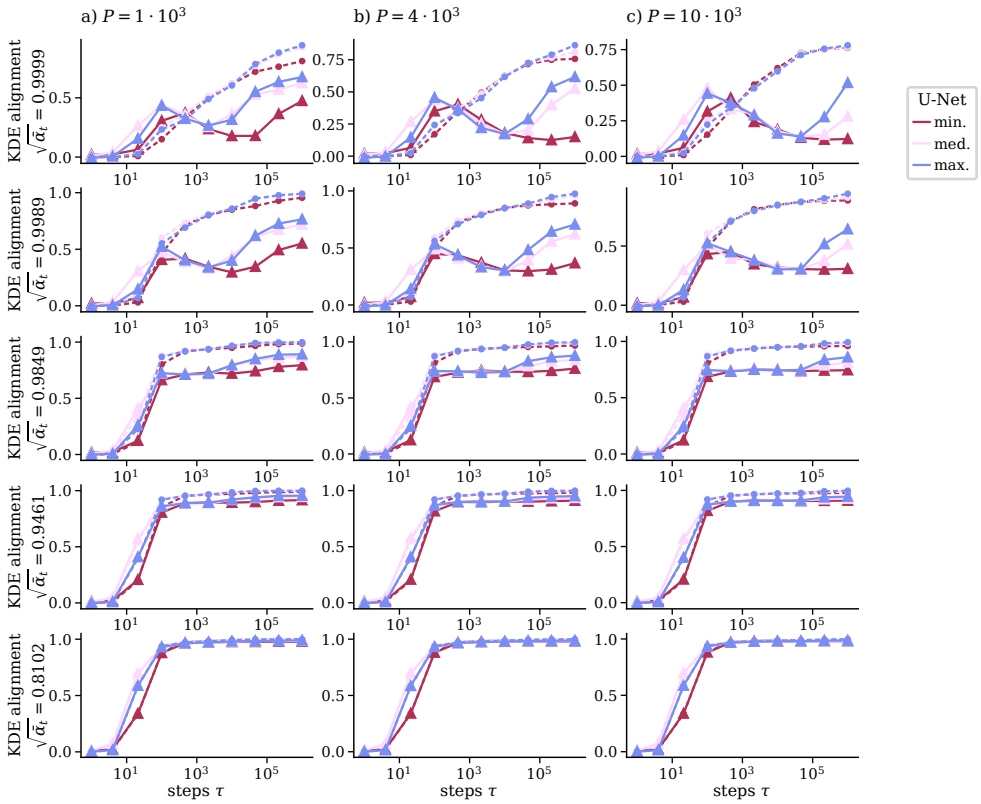


Figure 8: Cosine similarity of trained diffusion model outputs to KDE scores, as a function of training steps, computed on 4000 noised training and test examples (if the training set is smaller than 4000, we resample training data randomly and add a different noise realization according to the diffusion forward process). Left to right: increasing training set size $P = 1,000, 4,000, 10,000$. Top to bottom: increasing level of noise.

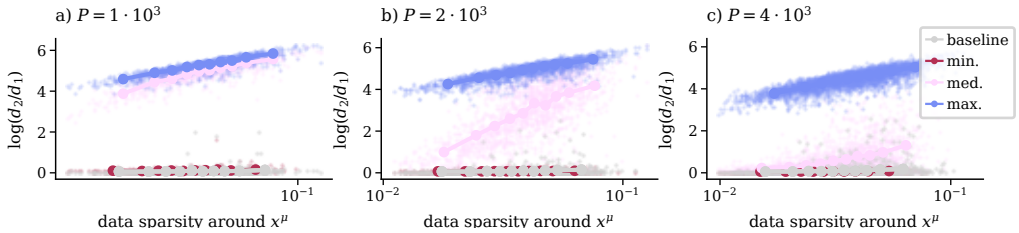


Figure 9: Local dominance of training data on generation for diffusion models using different U-Nets and different training datasets from CIFAR-10 with increasing sizes P. We generated $40,000$ samples per U-Net, each trained for 10^6 steps. Grey dots show the baseline for test data instead of generated data.

study. We compare generated samples to training and test data using either cosine similarity or normalized ℓ_2 distance. In the cosine case, all inputs are flattened and ℓ_2-normalized before computing similarities. Memorization is quantified at the level of individual generated samples. For each generated sample $\tilde{x}$, we compute its two nearest training neighbors and define distances d_1 and d_2. A sample is considered memorized if $\frac{d_1}{d_2} < \tau$ for a range of thresholds $\tau \in \{1/6, 1/4, 1/3, 1/2, 2/3\}$. This yields both per-sample and per-training-point memorization statistics. Local data density is estimated using nearest-neighbor statistics computed solely on the training set. For each training sample, we compute the distances to the k nearest neighbors for $k \in \{2, 5, 10, 20, 50\}$ and average over these distances as defined

in eq. (16). These statistics serve as proxies for local sparsity and are later correlated with memorization behavior.

D.1 CIFAR-10 Experiments

We construct the CIFAR-10 dataset by combining the original training and test splits (60,000 images total), and then randomly partitioning them into three equal subsets (train/validation/test), each containing approximately 20,000 images. All images are represented as 32×32 RGB tensors normalized to $[0, 1]$. No additional preprocessing or augmentation is applied.

D.2 MNIST + CIFAR-10 Experiments

To study memorization under controlled differences in local data density, we construct a combined dataset from CIFAR-10 and MNIST, which exhibit markedly different intrinsic complexities. CIFAR-10 images are highly variable, while MNIST digits occupy a much more concentrated region of image space. This setup allows us to induce systematic differences in local sparsity across classes.

We merge the original training and test splits of both datasets. To ensure compatibility, MNIST images are resized to 32×32 and converted to three channels by duplicating the grayscale channel, resulting in RGB tensors consistent with CIFAR-10. CIFAR-10 images are used without modification. All images are represented as 32×32 RGB tensors normalized to $[0, 1]$, and no additional preprocessing or augmentation is applied.

To isolate dataset-level effects, we ignore the original class labels and instead assign a binary label indicating the dataset of origin (CIFAR-10 or MNIST). The two datasets are then balanced by subsampling the larger dataset such that both contribute an equal number of samples. The resulting dataset therefore consists of an equal mixture of CIFAR-10 and MNIST images.

D.3 CelebA Experiments

This dataset consists of high-resolution face images together with 40 binary attributes per image. All images are resized to 32×32 pixels, and converted to grayscale. The resulting images are represented as single-channel tensors normalized to $[0, 1]$. No additional data augmentation is applied.

AI use statement

In this work, we used generative AI tools to aid and polish writing. We have not used generative AI tools to help develop theoretical models or conceptual frameworks, generate synthetic data sets,formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, assist with translation, clean and reformat datasets, support qualitative and thematic data analysis, interpret results. We have reviewed all AI-assisted work, by examining and correcting formulations and ascertaining that all writing improvements are correct. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.