Journal of Machine Learning Research 27 (2026) 1-50 Submitted 7/25; Revised 4/26; Published 6/26

Breaking the Curse of Dimensionality: Diffusion Models Efficiently Learn Low-Dimensional Distributions

Peng Wang[§,*] PENGW@UM.EDU.MO
Huijie Zhang[†,*] HUIJIEZH@UMICH.EDU
Zekai Zhang[†] ZZEKAI@UMICH.EDU
Siyi Chen[†] SIYICH@UMICH.EDU
Yi Ma[‡] YIMA@EECS.BERKELEY.EDU
Qing Qu[†] QINGQU@UMICH.EDU

[§] *Department of Computer and Information Science, University of Macau, Macao*
[†] *Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor*
[‡] *Department of Electrical Engineering and Computer Science, University of California, Berkeley*
[*] *The first and second authors contributed equally.*

Editor: Qiang Liu

Abstract

Despite their empirical success across a wide range of generative tasks, the fundamental principles underlying the ability of diffusion models to learn data distributions are poorly understood. In this work, we develop a new mathematical framework that explains how diffusion models can effectively learn low-dimensional distributions from a finite number of training samples without suffering from the curse of dimensionality. Specifically, motivated by the intrinsic low-dimensional structure of image data, we theoretically analyze a setting in which the data distribution is modeled as a mixture of low-rank Gaussians. Under a suitable network parameterization, we show that optimizing the training objective of diffusion models is equivalent to solving the canonical subspace clustering problem over the training samples, where each subspace basis corresponds to the low-rank covariance of a Gaussian component. This equivalence allows us to show that the sample complexity for learning the underlying distribution scales linearly with the intrinsic dimension of the data, rather than exponentially with the ambient dimension. Our theoretical findings are further supported by empirical evidence that demonstrates phase transition phenomena in generalization on both synthetic and real-world image datasets. Moreover, we establish a correspondence between the learned subspace bases and semantic attributes of image data, providing a principled foundation for controllable image generation.

Keywords: diffusion models, mixture of low-rank Gaussians, principal component analysis, subspace clustering, curse of dimensionality

1. Introduction

Generative modeling is a fundamental task in deep learning that seeks to learn the underlying data distribution from training samples to generate new and realistic data. Among recent advances, diffusion models have emerged as a powerful class of generative models,

achieving remarkable performance across a wide range of domains, including image generation (Ho et al., 2020; Wang et al., 2023b), video synthesis (Bar-Tal et al., 2024; Xing et al., 2024), speech and audio generation (Kong et al., 2020, 2021), and solving inverse problems (Fabian et al., 2024; Chung et al., 2023). In general, diffusion models learn a data distribution from training samples through a process that imitates the nonequilibrium thermodynamic diffusion process (Ho et al., 2020; Sohl-Dickstein et al., 2015; Song et al., 2021a). Specifically, a diffusion model operates in two stages: (i) a forward process, in which Gaussian noise is gradually added to the training data over a sequence of time steps; and (ii) a reverse process, in which the noise is progressively removed by a neural network trained to approximate the score function, i.e., the gradient of the log probability density function (pdf) of the data, at each time step.

Despite the great empirical success of diffusion models and recent advances in understanding their sampling convergence (Benton et al., 2024; Lee et al., 2022; Li and Cai, 2026; Li et al., 2024a, 2025b), distribution approximation (Chen et al., 2023b; Oko et al., 2023; Wang and Vastola, 2024), memorization (Gu et al., 2025; Somepalli et al., 2023a; Wen et al., 2023), and generalization (Han et al., 2024; Kadkhodaie et al., 2024; Zhang et al., 2024), the mechanisms underlying their performance remain poorly understood. This is primarily due to the black-box nature of neural networks and the inaccessibility of real-world data distributions. In particular, a fundamental question arises: Can diffusion models truly learn the underlying data distribution? If so, how many samples are required to achieve this? Recent theoretical studies (Oko et al., 2023; Wibisono et al., 2024) have shown that learning an arbitrary probability distribution using diffusion models inevitably suffers from the curse of dimensionality. Specifically, if the underlying density belongs to a generic class of probability distributions, obtaining an ϵ-accurate estimate of the corresponding score function requires the number of training samples that scales as $O(\epsilon^{-n})$, where n is the ambient data dimension. However, recent empirical studies (Zhang et al., 2024; Kadkhodaie et al., 2024) have shown that diffusion models can effectively learn image data distributions and generate novel and semantically meaningful samples distinct from the training data, even when trained on far fewer samples than those suggested by the existing theoretical bounds. As such, the gap between theory and practice raises a key question:

When and why can diffusion models learn data distributions
without suffering from the curse of dimensionality?

1.1 Our Contributions

In this work, we address the above question by investigating how diffusion models learn data distributions with intrinsic low-dimensional structures. Unlike previous studies (Oko et al., 2023; Wibisono et al., 2024), which considered arbitrary distributions, our study focuses on low-dimensional distributions, motivated by the observation that real-world image data often lie on a union of low-dimensional manifolds despite their high ambient dimensions (Brown et al., 2023; Kamkari et al., 2024; Loaiza-Ganem et al., 2024). These structures arise from underlying symmetries, repetitive patterns, and local regularities in natural images, which reduce the degrees of freedom in the data (Gong et al., 2019; Pope et al., 2020). To effectively capture the low-dimensional structure of real-world image data while offering analytical tractability, we focus on a mixture of low-rank Gaussians (`MoLRG`; see Defini-

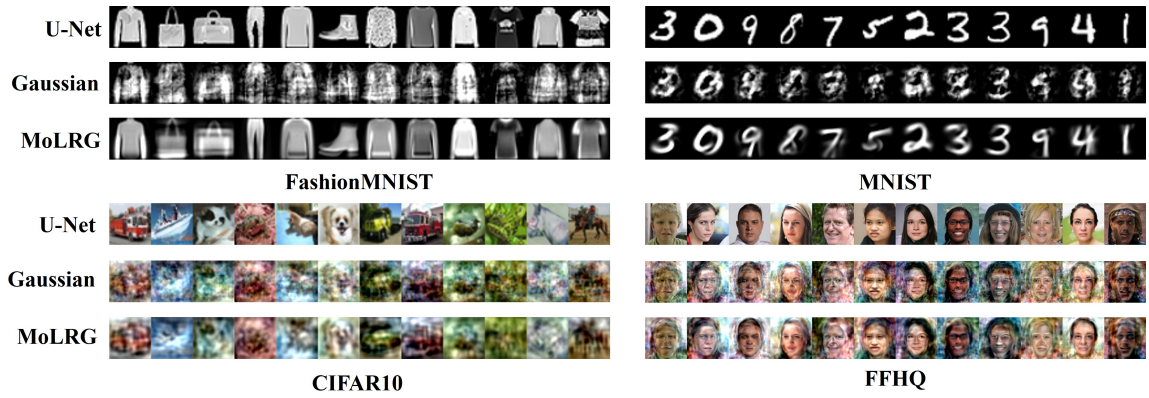


Figure 1: **Comparison of images generated from the Gaussian, `MoLRG`, and the distribution learned by diffusion models across different datasets.** Each row displays images generated from different distributions using the reverse-time ODE sampler, including the Gaussian, `MoLRG`, and the distribution learned by U-Net. The columns represent images generated from the same initial noise. The results are shown for four datasets: FashionMNIST (top left), MNIST (top right), CIFAR-10 (bottom left), and FFHQ (bottom right).

tion 1). Notably, our focus is further supported by empirical evidence in Figure 1, which demonstrates that samples generated via the diffusion reverse sampling process—using our theoretically constructed model—closely resemble those produced by U-Net (Ronneberger et al., 2015) trained on the same dataset and initialized with the same noise.

Theoretically, we show that diffusion models can learn the `MoLRG` distribution, provided that the minimum number of training samples scales linearly with the intrinsic dimension of the data, thereby overcoming the curse of dimensionality. Our result is established by demonstrating the equivalence between the training loss of diffusion models and the canonical subspace clustering problem (Vidal, 2011; Wang et al., 2022) (see Theorem 7) under an appropriate parameterization for the denoising autoencoder. Suppose that each component in the `MoLRG` distribution has zero mean and covariance matrix $\boldsymbol{U}_k\boldsymbol{U}_k^T$, where each low-rank matrix $\boldsymbol{U}_k$ has orthonormal columns and the subspaces spanned by $\{\boldsymbol{U}_k\}$ are mutually orthogonal. Notably, these assumptions imply that the data are distributed around a union of low-dimensional linear subspaces, with isotropic variation within each subspace and no cross-subspace correlations. Here, the orthogonality assumption serves as an idealized yet analytically tractable abstraction that captures well-separated components in high-dimensional spaces. Under these assumptions, our theory demonstrates a *phase transition* in the ability of diffusion models to learn the underlying distributions, which occurs when the number of training samples exceeds the intrinsic dimension of the data-generating subspaces (see Theorem 8). Moreover, our theoretical analysis offers valuable practical insights, as highlighted below.

- *The phase transition of generalization on image datasets.* As shown in Section 5.1, when training diffusion models on real-world image datasets, we observe a similar phase transition from failure to success in generalization, where the model begins to generate new

and sensible images distinct from the training data once the number of training samples exceeds a threshold that scales linearly with the intrinsic dimension of the data. Our study of `MoLRG` offers key insights into understanding this phenomenon.

- *Correspondence between subspace bases and semantic task vectors.*[1] We find that the basis vectors of the subspaces identified through our theoretical analysis align with semantically meaningful directions, that is, task vectors, in diffusion models pretrained on real-world image datasets. These semantic task vectors enable control over attributes such as gender, hairstyle, and color in the generated images (see Figure 5). This insight has inspired new training-free image editing methods on pretrained diffusion models (Chen et al., 2024).

Our study of distribution learning is closely related to recent studies on the generalization of diffusion models. It is well understood that when generative models successfully learn the true underlying data distribution, they exhibit strong generalization capabilities, enabling them to generate new samples that differ from the training data (Arora et al., 2017; Li et al., 2024b; Kadkhodaie et al., 2024). Moreover, recent empirical studies (Zhang et al., 2024; Kadkhodaie et al., 2024) have shown that strong generalization in diffusion models often corresponds to an accurate approximation of the underlying distribution, as evidenced by reproducibility. Specifically, these studies observed that different diffusion models can reproduce each other's outputs while generating new samples distinct from the training data, even when trained with different architectures, loss functions, and non-overlapping subsets of the training data. Motivated by these discussions, this work considers generalization in diffusion models as their ability to accurately capture the underlying data distribution. In this sense, our work also contributes to the theoretical understanding of generalization by characterizing the sample complexity required for diffusion models to learn the underlying distribution.

1.2 Notation and Organization

Notation. We write matrices in bold capital letters, such as $\boldsymbol{A}$, vectors in bold lower-case letters, such as $\boldsymbol{a}$, and scalars in plain letters, such as a. Given a matrix $\boldsymbol{A}$, we use $\|\boldsymbol{A}\|$ to denote its largest singular value (i.e., spectral norm), $\sigma_i(\boldsymbol{A})$ its i-th largest singular value, a_{ij} its (i,j)-th entry, $\mathrm{rank}(\boldsymbol{A})$ its rank, and $\|\boldsymbol{A}\|_F$ its Frobenius norm. Given a vector $\boldsymbol{a}$, we use $\|\boldsymbol{a}\|$ to denote its Euclidean norm and a_i to denote its i-th entry. Let $\mathcal{O}^{n\times d}$ denote the set of all $n \times d$ matrices that have orthogonal columns. We simply write the score function $\nabla_{\boldsymbol{x}} \log p(\boldsymbol{x})$ of a distribution with pdf $p(\boldsymbol{x})$ as $\nabla \log p(\boldsymbol{x})$. We denote by $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ a multivariate Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma} \succeq \boldsymbol{0}$.

Organization. In Section 2, we introduce the preliminaries of diffusion models and state our assumptions regarding the data and model. In Section 3, we present the main results of this study. In Section 4, we discuss how our results relate to the existing literature. In Section 5, we conduct numerical experiments to support our theory and demonstrate its practical implications. Finally, in Section 6, we summarize our work and discuss potential directions for future research. All proofs are presented in the appendices.

1. A semantic task vector is a direction in the latent or intermediate feature space such that traversing along it causes a controlled and interpretable change in the output.

2. Problem Setup

In this section, we introduce preliminaries on diffusion models and state our assumptions on the data and model. We consider a training dataset $\{\boldsymbol{x}^{(i)}\}_{i=1}^N \subseteq \mathbb{R}^n$, where the data points are independently and identically distributed (*i.i.d.*) samples from the underlying data distribution $p_{\text{data}}(\boldsymbol{x})$; that is, $\boldsymbol{x}^{(i)} \overset{i.i.d.}{\sim} p_{\text{data}}(\boldsymbol{x})$.

2.1 Preliminaries on Score-Based Diffusion Models

Forward and reverse processes of diffusion models. In general, diffusion models aim to learn a data distribution and generate new samples through forward and reverse processes indexed by a continuous time variable $t \in [0,1]$. Specifically, the forward process progressively injects noise into the data, which can be described by the following stochastic differential equation (SDE):

$$\mathrm{d}\boldsymbol{x}_t = f(t)\boldsymbol{x}_t \mathrm{d}t + g(t)\mathrm{d}\boldsymbol{w}_t, \tag{1}$$

where $\boldsymbol{x}_0 \sim p_{\text{data}}(\boldsymbol{x})$, scalar functions $f(t), g(t) : [0,1] \to \mathbb{R}$ denote the drift and diffusion coefficients, respectively, and $\{\boldsymbol{w}_t\}_{t\in[0,1]}$ is the standard Wiener process. For ease of exposition, let $p_t(\boldsymbol{x})$ denote the *pdf* of $\boldsymbol{x}_t$ and $p_t(\boldsymbol{x}_t \mid \boldsymbol{x}_0)$ be the transition kernel from $\boldsymbol{x}_0$ to $\boldsymbol{x}_t$.[2] According to (1), one can verify that

$$p_t(\boldsymbol{x}_t \mid \boldsymbol{x}_0) = \mathcal{N}\left(\boldsymbol{x}_t; s_t\boldsymbol{x}_0, s_t^2\sigma_t^2\boldsymbol{I}_n\right), \tag{2}$$

where $s_t := \exp\left(\int_0^t f(\xi)\mathrm{d}\xi\right)$ and $\sigma_t := \sqrt{\int_0^t g^2(\xi)/s^2(\xi)\mathrm{d}\xi}$.[3]

The reverse process gradually removes the noise from $\boldsymbol{x}_1$ using the following probability-flow ordinary differential equation (ODE) backward in time:

$$\mathrm{d}\boldsymbol{x}_t = \left(f(t)\boldsymbol{x}_t - \frac{1}{2}g^2(t)\nabla \log p_t(\boldsymbol{x}_t)\right)\mathrm{d}t. \tag{3}$$

If $\boldsymbol{x}_1 \sim p_1$ and the score function $\nabla \log p_t(\boldsymbol{x}_t)$ is known for all $t \in [0,1]$, then this reverse-time ODE has the same marginal distribution p_t as the forward process at each time $t \in [0,1]$ (Song et al., 2021b).

Training loss of diffusion models. Unfortunately, the score function $\nabla \log p_t$ at each $t \in [0,1]$ is typically unknown, as the marginal distribution p_t is induced by the unknown data distribution p_{data}. To enable data generation via the reverse-time ODE in (3), we train a neural network to approximate the score function from the training data. In addition, Tweedie's formula (Efron, 2011) relates the score function $\nabla \log p_t(\boldsymbol{x}_t)$ to the posterior mean $\mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t]$ as follows:

$$s_t\mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t] = \boldsymbol{x}_t + s_t^2\sigma_t^2\nabla \log p_t(\boldsymbol{x}_t), \ \forall t \in (0,1]. \tag{4}$$

This allows us to estimate the posterior mean $\mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t]$ as an alternative approach for estimating the score function $\nabla \log p_t(\boldsymbol{x}_t)$. Leveraging the strong function approximation

2. Note that $p_0 := p_{\text{data}}$.
3. With a slight abuse of notation, we denote $s_t := s(t)$ and $\sigma_t := \sigma(t)$.

capabilities of neural networks (Hornik et al., 1989), recent studies (Chen et al., 2025d; Kadkhodaie et al., 2024; Karras et al., 2022; Wang et al., 2023a; Vincent, 2011) have explored training a time-dependent neural network $\boldsymbol{x}_{\boldsymbol{\theta}}(\cdot, t) : \mathbb{R}^n \times [0,1] \to \mathbb{R}^n$ with parameters $\boldsymbol{\theta}$, referred to as the *denoising autoencoder* (DAE), to estimate the posterior mean $\mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t]$. To learn the network parameters $\boldsymbol{\theta}$, we minimize the following empirical loss over the training samples $\{\boldsymbol{x}^{(i)}\}_{i=1}^N$:

$$\min_{\boldsymbol{\theta}} \ \ell(\boldsymbol{\theta}) = \frac{1}{N}\sum_{i=1}^{N}\int_0^1 \lambda_t \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(\boldsymbol{0},\boldsymbol{I}_n)}\left[\left\|\boldsymbol{x}_{\boldsymbol{\theta}}(s_t\boldsymbol{x}^{(i)} + \gamma_t\boldsymbol{\epsilon}, t) - \boldsymbol{x}^{(i)}\right\|^2\right] \mathrm{d}t, \tag{5}$$

where $\gamma_t := s_t\sigma_t$ and $\lambda_t : [0,1] \to \mathbb{R}^+$ is the weighting function.

2.2 Mixture of Low-Rank Gaussians

In this work, we consider learning a mixture of low-rank Gaussians (`MoLRG`), which effectively captures the intrinsic low-dimensional structure of real-world image datasets while maintaining analytical tractability. Specifically, the `MoLRG` distribution is defined as follows.

Definition 1 (Mixture of Low-Rank Gaussians) *We say that a random vector $\boldsymbol{x}_0 \in \mathbb{R}^n$ follows a mixture of K low-rank Gaussian distributions with mixing proportions $\{\pi_k\}_{k=1}^K$, means $\{\boldsymbol{\mu}_k^\star\}_{k=1}^K \subseteq \mathbb{R}^n$, and covariance matrices $\{\boldsymbol{\Sigma}_k^\star\}_{k=1}^K \subseteq \mathbb{R}^{n\times n}$ if its distribution is given by*

$$\boldsymbol{x}_0 \sim \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{\mu}_k^\star, \boldsymbol{\Sigma}_k^\star), \tag{6}$$

where $\pi_k \geq 0$ is the mixing proportion of the k-th component satisfying $\sum_{k=1}^K \pi_k = 1$, and $\boldsymbol{\mu}_k^\star$ and $\boldsymbol{\Sigma}_k^\star \succeq \boldsymbol{0}$ denote the mean and covariance matrix of the k-th component, respectively. In particular, the covariance matrix $\boldsymbol{\Sigma}_k^\star$ is low-rank with $\mathrm{rank}(\boldsymbol{\Sigma}_k^\star) = d_k < n$.

Remarks. Intuitively, data drawn from a `MoLRG` distribution lie on a union of low-dimensional linear subspaces, where the k-th subspace is characterized by the mean $\boldsymbol{\mu}_k^\star$ and low-rank covariance matrix $\boldsymbol{\Sigma}_k^\star$. We now discuss the motivation for studying this model and its connections to other distributions that have been theoretically analyzed.

- *`MoLRG` captures the low-dimensional structure of real-world image datasets.* Recent studies (Brown et al., 2023; Kamkari et al., 2024) conducted extensive numerical experiments and demonstrated that image datasets, such as MNIST (LeCun et al., 1998), CIFAR-10 (Krizhevsky et al., 2009), and ImageNet (Russakovsky et al., 2015), approximately reside on a union of low-dimensional manifolds. Locally, each nonlinear manifold can be effectively approximated using its tangent space (i.e., a linear subspace). Consequently, the `MoLRG` model, which represents the data as a union of linear subspaces, provides a suitable local approximation for real-world image data distributions. This claim is supported by the empirical studies in Section 2.4. In addition, the latent distribution of real-world data can be well approximated by a Gaussian, as modern diffusion models typically employ autoencoders with KL regularization to encourage alignment with a standard Gaussian prior (Kingma and Welling, 2014; Rombach et al., 2022). This latent

Gaussian structure, as adopted in the MoLRG model, also facilitates theoretical analysis, allowing us to derive the closed-form expression for the posterior estimator at each time step, as shown in Lemma 10. Therefore, studying the MoLRG model is a valuable starting point for theoretical studies on the distribution learning capability of diffusion models.

- *Comparison with recent studies on a mixture of Gaussians.* Many recent studies have investigated how diffusion models learn a mixture of *isotropic* Gaussians (MoG), that is, $\boldsymbol{\Sigma}_k^\star = \boldsymbol{I}_n$ in (6); see, e.g., Chen et al. (2025b); Cole and Lu (2024); Gatmiry et al. (2025); Shah et al. (2023); Wu et al. (2024). These studies mainly focus on learning the means of the Gaussian components, provided that each covariance matrix is fixed as the identity. In contrast, our work considers a mixture of *low-rank* Gaussians, where the key challenge lies in learning the low-rank covariance matrices instead of the means. The low-rankness captures the inherent low-dimensionality of image datasets (Gong et al., 2019; Pope et al., 2020; Stanczuk et al., 2024) and offers deeper insight into why diffusion models learn data distributions in practice without suffering from the curse of dimensionality. In addition, several studies have investigated the reverse sampling process of diffusion models based on a mixture of Gaussians. For example, Biroli et al. (2024) analyzed a mixture of two Gaussians with distinct means and identical variance, revealing that the reverse diffusion process exhibits distinct dynamical regimes. In addition, Li et al. (2025a) demonstrated that diffusion models can efficiently sample from high-dimensional distributions that are well approximated by a mixture of Gaussians. In comparison, our work focuses on the training process of diffusion models rather than the sampling process.

  Additionally, a single Gaussian, as a special case of a mixture of Gaussians, has been extensively studied owing to its analytical tractability, despite its limited expressive power. For example, Wang and Vastola (2023, 2024); Li et al. (2024c) empirically demonstrated that the score function of a well-trained diffusion model at a high-noise scale is well approximated by the score of a single Gaussian. In addition, Chen et al. (2025c) uses a single Gaussian model to show that denoising score distillation can identify the eigenspace of the covariance matrix of a Gaussian.

2.3 Network Parameterization Inspired by MoLRG

To analyze the distribution learning behavior of diffusion models, one natural approach is to study the training loss in (5). This approach critically depends on a suitable parameterization of the DAE $\boldsymbol{x}_{\boldsymbol{\theta}}(\cdot, t)$. In practice, $\boldsymbol{x}_{\boldsymbol{\theta}}(\cdot, t)$ is typically parameterized by a U-Net architecture (Ronneberger et al., 2015), which consists of deep nonlinear encoder and decoder networks with skip connections. However, the highly nonlinear structure of U-Net poses significant challenges for theoretical analysis.

To enable analytical tractability while retaining structural similarity to U-Net, we consider a network architecture for $\boldsymbol{x}_{\boldsymbol{\theta}}(\cdot, t)$ that is a combination of multiple one-layer linear encoders and decoders, weighted by a softmax-like function, as follows:

$$\boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) = \sum_{k=1}^{K} w_{k,t}(\boldsymbol{x}_t) \left(\boldsymbol{\mu}_k + \boldsymbol{U}_k \boldsymbol{D}_{k,t} \boldsymbol{U}_k^T \left(\frac{\boldsymbol{x}_t}{s_t} - \boldsymbol{\mu}_k \right) \right), \tag{7}$$

where $\boldsymbol{\theta} = \{(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{U}_k, \boldsymbol{\Lambda}_k)\}_{k=1}^K$ denotes the network parameters, $\boldsymbol{U}_k \in \mathcal{O}^{n\times d_k}$ has orthonormal columns, and $\boldsymbol{\Lambda}_k = \mathrm{diag}(\lambda_{k,1}, \ldots, \lambda_{k,d_k})$ is a diagonal matrix. Additionally, $\boldsymbol{D}_{k,t}$ and $w_{k,t}(\boldsymbol{x}_t)$ are defined as follows:

$$\boldsymbol{D}_{k,t} = \mathrm{diag}\left(\frac{s_t^2\lambda_{k,1}}{\gamma_t^2 + s_t^2\lambda_{k,1}}, \ldots, \frac{s_t^2\lambda_{k,d_k}}{\gamma_t^2 + s_t^2\lambda_{k,d_k}}\right),\ w_{k,t}(\boldsymbol{x}) = \frac{\pi_k \mathcal{N}\left(\boldsymbol{x}; s_t\boldsymbol{\mu}_k, s_t^2\boldsymbol{U}_k\boldsymbol{\Lambda}_k\boldsymbol{U}_k^T + \gamma_t^2\boldsymbol{I}_n\right)}{\sum_{l=1}^K \pi_l \mathcal{N}\left(\boldsymbol{x}; s_t\boldsymbol{\mu}_l, s_t^2\boldsymbol{U}_l\boldsymbol{\Lambda}_l\boldsymbol{U}_l^T + \gamma_t^2\boldsymbol{I}_n\right)},$$

where σ_t and γ_t are defined in Section 2.1. Our network architecture in (7) can be viewed as a mixture-of-experts architecture (Shazeer et al., 2017), where each expert network consists of a linear encoder $\boldsymbol{U}_k^T$ and decoder $\boldsymbol{U}_k$. These experts are then combined through a learnable weighted summation, allowing the model to adaptively assign weights among components. In addition to the resemblance to U-Net, the parameterization in (7) is well motivated from the following perspectives:

- *Inspired by the posterior mean of* `MoLRG`. As the DAE serves as an estimator of the posterior mean (i.e., $\boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) \approx \mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t]$), our network parameterization is inspired by the analytical form of the posterior mean $\mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t]$ of `MoLRG`; see Lemma 10 in Appendix A. Note that the network parameters $\boldsymbol{\theta}$ in (7) are learnable instead of being the ground-truth of the means and covariances of the `MoLRG`. In Section 3, we will investigate how to learn the network parameters in simplified settings.
- *Meaningful image generation through the parameterization.* When the network parameters $\boldsymbol{\theta}$ are directly estimated from training data, the experimental results in Section 2.4 demonstrate that our parameterization (7) generates images that are coarsely similar to those produced by a standard U-Net. This demonstrates the practical effectiveness of the proposed parameterization on real-world datasets and supports its ability to capture coarse image structure. Further details are provided below.

2.4 Experimental Support for Data and Model Assumptions

As illustrated in Figure 1 and Table 1, we empirically validate the `MoLRG` assumption and the corresponding network parameterization introduced in (7) for approximating real-world data distributions. In our experiments, we used the distribution learned by U-Net as a benchmark. To quantify the similarity between the images generated by (7) and those produced by U-Net, we computed the following metric:

$$\frac{1}{M}\sum_{i=1}^{M}\left\|\boldsymbol{y}_1^{(i)} - \boldsymbol{y}_2^{(i)}\right\|, \tag{8}$$

where M denotes the number of generated samples, and $\boldsymbol{y}_1^{(i)}$ and $\boldsymbol{y}_2^{(i)}$ denote the i-th samples generated from the distributions learned by U-Net and the parameterization in (7), respectively. Here, both sets of samples are generated using the reverse-time ODE in (3), initialized with the same noise. Detailed experimental setups are provided in Appendix C.

Based on the above experimental setup, we conducted experiments on real-world image datasets, including MNIST (Deng, 2012), FashionMNIST (Xiao et al., 2017), CIFAR-10 (Krizhevsky et al., 2009), and FFHQ (Kazemi and Sullivan, 2014). Our proposed model and network architecture were then compared against existing approaches.

	FashionMNIST	MNIST	CIFAR-10	FFHQ
Gaussian	72.62	69.12	33.55	36.75
`MoLRG`	**57.56**	**62.53**	**31.29**	**35.78**

Table 1: Distance (defined in (8)) between samples generated from the theoretical network parameterization (based on MoLRG or Gaussian) and those generated by U-Net.

- *Comparison between our model and U-Net.* First, we compare images generated by U-Net trained on the real-world dataset $\{\boldsymbol{x}^{(i)}\}_{i=1}^{N}$ with those generated by our parameterized network in (7), which uses the means and covariances estimated from the same dataset. As illustrated in Figure 1, images generated by the two network parameterizations using the same sampling procedure exhibit substantial visual similarity, especially on simpler datasets such as FashionMNIST and MNIST. This observation supports the validity of our data assumptions and confirms the effectiveness of our network parameterization in approximating real-world distributions. On more complex datasets, such as CIFAR-10 and FFHQ, although our parameterized network cannot capture fine-grained image details, it preserves the overall structural characteristics of the images generated by U-Net. The loss of fine details indicates a limitation of our model assumptions, which merits further investigation.

- *Comparison between our model and the single full-rank Gaussian parameterization.* In addition, we compared our model with a network parameterized according to a single full-rank Gaussian model, as explored in prior studies (Wang and Vastola, 2024; Li et al., 2024c). As illustrated in Figure 1, single Gaussian parameterization often results in high intra-class variance and blurred images, particularly on simpler datasets such as FashionMNIST and MNIST (second row of the figure). In contrast, our model based on `MoLRG` significantly improves generation quality (third row) by leveraging multiple mixture components to mitigate intra-class variance and employing low-rank covariance structures to suppress high-frequency noise. Moreover, as shown in Table 1, despite employing fewer parameters, our model consistently outperforms the single Gaussian model in terms of the distance to images generated by U-Net.

3. Sample Complexity Analysis for Learning `MoLRG`

Building upon the setup introduced in Section 2, we theoretically analyze the sample complexity of learning the `MoLRG` distribution via diffusion models. Specifically, we show that

- The training loss of diffusion models in (5) under our parameterization is equivalent to the canonical subspace clustering problem.
- The minimum number of samples required for learning `MoLRG` via diffusion models scales linearly with the intrinsic data dimension.

Because real-world images often contain noise owing to sensor imperfections or environmental conditions, we additionally incorporate an additive noise term into the `MoLRG` model.

Consequently, we formalize the noisy data generation process under the `MoLRG` model as follows:

Assumption 2 *For each $i \in [N]$, let $z_i \in [K]$ be a latent component label with $\mathbb{P}(z_i = k) = \pi_k$. Conditional on $z_i = k$, the training sample is generated according to*

$$\boldsymbol{x}^{(i)} = \boldsymbol{U}_k^\star \boldsymbol{a}_i + \boldsymbol{e}_i, \tag{9}$$

where $\boldsymbol{a}_i \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_{d_k})$ denotes the latent coefficient vector, $\boldsymbol{U}_k^\star \in \mathbb{R}^{n \times d_k}$ has orthonormal columns, and $\boldsymbol{e}_i \in \mathbb{R}^n$ denotes an additive noise vector.[4]

Notably, because the `MoLRG` distribution is fully characterized by the first- and second-order moments of each degenerate Gaussian component, learning this distribution reduces to estimating the bases $\{\boldsymbol{U}_k^\star\}_{k=1}^K$ according to our setup. In the following, we demonstrate that this estimation can be achieved by minimizing the DAE training loss in Problem (5) with respect to optimization variables $\{\boldsymbol{U}_k\}_{k=1}^K$.

3.1 A Warm-Up Study: Learning a Single Low-Rank Gaussian

To build intuition, we begin by introducing our result in a simple setting, where the underlying distribution p_{data} is a *single* low-rank Gaussian, i.e., $K = 1$ in (9). Specifically, the training samples $\{\boldsymbol{x}^{(i)}\}_{i=1}^N$ are generated according to

$$\boldsymbol{x}^{(i)} = \boldsymbol{U}^\star \boldsymbol{a}_i + \boldsymbol{e}_i, \tag{10}$$

where $\boldsymbol{U}^\star \in \mathcal{O}^{n \times d}$ denotes an orthonormal basis, $\boldsymbol{a}_i \overset{i.i.d.}{\sim} \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_d)$ is the coefficient for each $i \in [N]$, and $\boldsymbol{e}_i \in \mathbb{R}^n$ is noise for all $i \in [N]$. According to (7), the parameterization of the DAE in this case reduces to

$$\boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) = \frac{s_t}{s_t^2 + \gamma_t^2} \boldsymbol{U}\boldsymbol{U}^T \boldsymbol{x}_t, \tag{11}$$

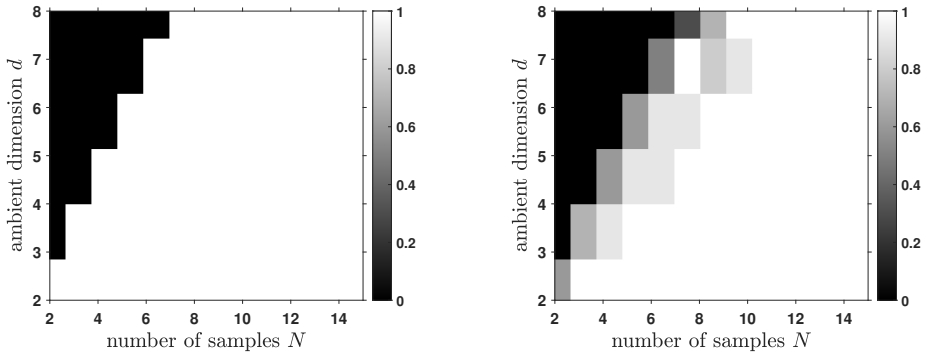
where $\boldsymbol{\theta} = \boldsymbol{U} \in \mathcal{O}^{n \times d}$. Equipped with the above setup, we obtain the following results.

Theorem 3 *Suppose that the DAE $\boldsymbol{x}_{\boldsymbol{\theta}}(\cdot, t)$ in Problem (5) is parameterized into* (11) *for each $t \in [0, 1]$. Then, Problem (5) is equivalent to the following principal component analysis (PCA) problem:*

$$\max_{\boldsymbol{U} \in \mathbb{R}^{n \times d}} \sum_{i=1}^N \left\| \boldsymbol{U}^T \boldsymbol{x}^{(i)} \right\|^2 \qquad \text{s.t.} \quad \boldsymbol{U}^T \boldsymbol{U} = \boldsymbol{I}_d. \tag{12}$$

We defer the proof to Appendix B.1. In this case, Theorem 3 shows that training diffusion models with the network parameterization (11) is equivalent to performing PCA on the training samples. Note that PCA is a classical and well-studied method for learning low-dimensional subspaces, whose optimal solution can be computed via singular value decomposition (SVD). This closed-form solution allows us to leverage existing results, such as Wedin's Theorem (Wedin, 1972), to facilitate our analysis. Consequently, we can apply classical tools to analyze the sample complexity of learning the underlying distribution with diffusion models as follows.

4. The signal component satisfies Definition 1, since $\boldsymbol{U}_k^\star \boldsymbol{a}_i \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{U}_k^\star \boldsymbol{U}_k^{\star T})$.



(a) **PCA** (b) **Diffusion Model**

Figure 2: **Phase transition of learning the `MoLRG` distribution with $K = 1$.** The x-axis is the number of training samples and y-axis is the dimension of subspaces. Darker pixels represent a lower empirical probability of success. We apply SVD and stochastic gradient descent to solve Problems (12) and (5), visualizing the results in (a) and (b), respectively.

Theorem 4 *Under the same setting as Theorem 3, suppose that Assumption 2 holds. The following statements hold:*

(i) If $N \geq d$, it holds with probability at least $1-1/2^{N-d+1}-\exp(-c_2 N)$ that any optimal solution $\hat{\boldsymbol{U}}$ satisfies

$$\left\|\hat{\boldsymbol{U}}\hat{\boldsymbol{U}}^T - \boldsymbol{U}^\star\boldsymbol{U}^{\star T}\right\|_F \leq \frac{c_1\sqrt{\sum_{i=1}^N \|\boldsymbol{e}_i\|^2}}{\sqrt{N}-\sqrt{d-1}}, \tag{13}$$

where $c_1, c_2 > 0$ are constants.

(ii) If $N < d$, there exists an optimal solution $\hat{\boldsymbol{U}} \in \mathcal{O}^{n\times d}$ such that with probability at least $1 - 1/2^{d-N+1} - \exp(-c_2' d)$,

$$\left\|\hat{\boldsymbol{U}}\hat{\boldsymbol{U}}^T - \boldsymbol{U}^\star\boldsymbol{U}^{\star T}\right\|_F \geq \alpha - \frac{c_1'\sqrt{\sum_{i=1}^N \|\boldsymbol{e}_i\|^2}}{\sqrt{d}-\sqrt{N-1}}, \tag{14}$$

where $\alpha := \sqrt{2\min\{d-N, n-d\}}$ and $c_1', c_2' > 0$ are constants.

We defer the proof to Appendix B.2. Next, we discuss the implications of our results.

- *Phase transition in learning the underlying distribution.* Building on the equivalence in Theorem 3 and the data model in (10), Theorem 4 clearly demonstrates a phase transition from failure to success of learning the underlying distribution via diffusion models as the number of training samples increases. More precisely, when the number of training samples is larger than the intrinsic dimension of the subspace, i.e., $N \geq d$, any optimal solution $\hat{\boldsymbol{U}}$ recovers the underlying subspace up to an approximation error determined by

the noise level. Conversely, when $N < d$, the training loss admits an optimal solution that fails to recover the underlying subspace. This phase transition is further corroborated by our experiments in Figure 2. Finally, because a Gaussian distribution can be fully characterized by its first two moments, our result rigorously shows that diffusion models can recover the underlying distribution when $N \geq d$, with the covariance estimation error bounded by the noise level.

- *The connection between PCA and semantic task vectors.* The correspondence between principal components and semantic meaning has been well studied in the machine learning literature. For example, early work (Turk and Pentland, 1991) demonstrated that PCA can reveal meaningful components of variation in natural image datasets, such as facial expression, lighting, and pose, suggesting a connection between directions of maximal variance and human-perceived semantics. Inspired by this insight, our empirical results in Section 5.2 reveal a similar phenomenon in diffusion models: task vectors can be identified through the leading singular vectors of the Jacobian of the DAE, which effectively capture distinct semantic features of natural images and enable controllable image generation.

3.2 Learning a Mixture of Low-Rank Gaussians

In this subsection, we extend the above study to the `MoLRG` distribution with $K > 1$. For ease of analysis, we impose the following additional assumption on Assumption 2:

Assumption 5 *Under Assumption 2, the subspace bases satisfy $\boldsymbol{U}_k^{\star T}\boldsymbol{U}_l^{\star} = \boldsymbol{0}$ for all $k \neq l$, the subspace dimensions are equal, i.e., $d_1 = \cdots = d_K = d$, and the mixing weights satisfy $\pi_1 = \cdots = \pi_K = 1/K$.*

This assumption enforces a symmetric and well-separated multi-subspace structure, which simplifies the analysis while preserving the essential geometric characteristics of the model. Moreover, we consider a hardmax counterpart of (7) for the DAE parameterization as follows:

Assumption 6 *The DAE is parameterized as*

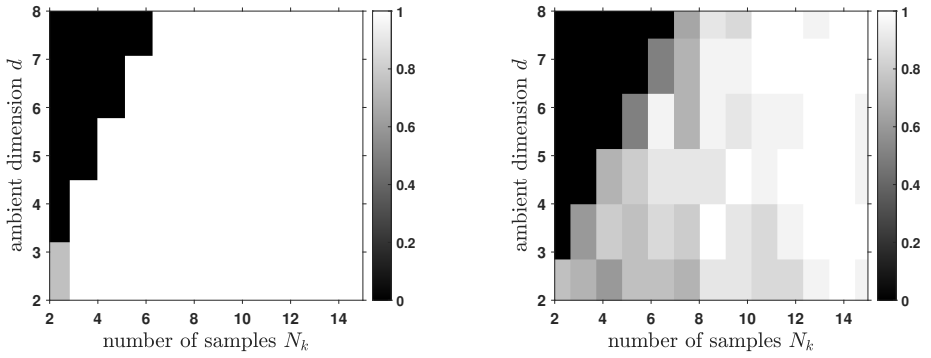
$$\boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) = \frac{s_t}{s_t^2 + \gamma_t^2} \sum_{k=1}^{K} \hat{w}_k(\boldsymbol{\theta}; \boldsymbol{x}_0) \boldsymbol{U}_k \boldsymbol{U}_k^T \boldsymbol{x}_t, \tag{15}$$

where $\boldsymbol{\theta} = \{\boldsymbol{U}_k\}_{k=1}^K$, $\boldsymbol{U} = [\boldsymbol{U}_1, \ldots, \boldsymbol{U}_K] \in \mathcal{O}^{n \times dK}$, and $\{\hat{w}_k(\boldsymbol{\theta}; \boldsymbol{x}_0)\}_{k=1}^K$ are defined as

$$\hat{w}_k(\boldsymbol{\theta}; \boldsymbol{x}_0) = \begin{cases} 1, & \text{if } k = k_0, \\ 0, & \text{otherwise}, \end{cases} \tag{16}$$

where $k_0 = \min\left\{j \in [K] : j \in \arg\max_{l \in [K]} \|\boldsymbol{U}_l^T \boldsymbol{x}_0\|\right\}$.

This assumption replaces the softmax weights in (7) with hardmax weights based on the clean training sample $\boldsymbol{x}_0$. This simplification facilitates the analysis while capturing the dominant-component behavior of the softmax weights. We refer the reader to Appendix B.3 for a discussion of how these hardmax weights approximate the softmax weights in (7). Now, we are ready to present the following theorem.



(a) **Subspace Clustering** (b) **Diffusion Model**

Figure 3: **Phase transition of learning the `MoLRG` distribution with $K = 2$.** The x-axis is the number of training samples and y-axis is the dimension of subspaces. Darker pixels represent a lower empirical probability of success. We apply a subspace clustering method and stochastic gradient descent to solve Problems (17) and (5), visualizing the results in (a) and (b), respectively. Additional experiments for the case when $K = 3$ are presented in Figure 6.

Theorem 7 *Suppose that Assumption 6 holds. Then, Problem* (5) *is equivalent to the following subspace clustering problem:*

$$\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{k=1}^{K} \sum_{i \in C_k(\boldsymbol{\theta})} \|\boldsymbol{U}_k^T \boldsymbol{x}^{(i)}\|^2 \qquad \text{s.t. } [\boldsymbol{U}_1, \ldots, \boldsymbol{U}_K] \in \mathcal{O}^{n \times dK}, \tag{17}$$

where

$$C_k(\boldsymbol{\theta}) := \left\{ i \in [N] : k = \min \left\{ j \in [K] : j \in \arg\max_{l \in [K]} \left\| \boldsymbol{U}_l^T \boldsymbol{x}^{(i)} \right\| \right\} \right\}, \ \forall k \in [K]. \tag{18}$$

We defer the proof of this theorem to Appendix B.4. In this theorem, the set $C_k(\boldsymbol{\theta})$ contains the data points assigned to the k-th learned subspace, where each data point is assigned to the subspace with the largest projection norm; in the case of ties, it is assigned to the one with the smallest index. Problem (17) seeks to maximize the sum of squared projection norms of data points onto their assigned subspaces. With the network parameterization in (15), Theorem 7 shows that optimizing the training loss of diffusion models is equivalent to solving the subspace clustering problem (Vidal, 2011; Wang et al., 2022). Notably, subspace clustering is a fundamental problem in unsupervised learning, which aims to identify and group data points that lie in a union of low-dimensional subspaces in a high-dimensional space (Vidal, 2011; Lerman and Maunu, 2018). By showing an equivalence between training diffusion models and subspace clustering in Theorem 7, we can characterize the minimum number of samples required for learning the underlying `MoLRG` distribution, similar to Theorem 4.

Before proceeding, we clarify the transition from mixing weights to component sample sizes. Each sample has a latent label $z_i \in [K]$ with $\mathbb{P}(z_i = k) = \pi_k$. We define

$$C_k^\star := \{i \in [N] : z_i = k\}, \quad N_k := |C_k^\star|.$$

Thus, π_k denotes the population-level mixing proportion, whereas N_k is the realized number of samples from the k-th component. Under Assumption 5, $(N_1, \ldots, N_K)$ follows a multinomial distribution with equal probabilities $1/K$, and $N_{\min} := \min_{k\in[K]} N_k$ is of order N/K with high probability. Therefore, the theorem below states the recovery guarantees in terms of the realized counts N_k, which determine the recoverability of each subspace.

Theorem 8 *Suppose that Assumptions 2, 5, and 6 hold, $d \gtrsim \log N$, and*

$$\max_{i\in[N]} \|\boldsymbol{e}_i\| \le c_\delta \min\left\{\frac{N_{\min}}{K^3 N\sqrt{d}}, \frac{\sqrt{d}}{N}\right\}, \tag{19}$$

where $c_\delta > 0$ is a sufficiently small absolute constant. Then the following statements hold:

(i) Suppose in addition that $N_{\min} \gtrsim K^6\big(d\log K + \log N\big)$. Then, with probability at least $1 - N^{-\Omega(1)}$, for any optimal solution $\{\hat{\boldsymbol{U}}_k\}_{k=1}^K$ of Problem (5), there exists a permutation $\Pi : [K] \to [K]$ such that, for all $k \in [K]$,

$$\left\|\hat{\boldsymbol{U}}_{\Pi(k)}\hat{\boldsymbol{U}}_{\Pi(k)}^T - \boldsymbol{U}_k^\star \boldsymbol{U}_k^{\star T}\right\|_F \le \frac{c_1\sqrt{\sum_{i\in C_k^\star}\|\boldsymbol{e}_i\|^2}}{\sqrt{N_k} - \sqrt{d-1}}, \tag{20}$$

where $c_1 > 0$ is a constant.

(ii) If $N_{k_0} < d$ for some $k_0 \in [K]$, then with probability at least $1 - 2^{-(d-N_{k_0}+1)} - \exp(-c_2' d)$, the PCA problem over the true cluster $C_{k_0}^\star$ admits an optimal solution $\tilde{\boldsymbol{U}}_{k_0} \in \mathcal{O}^{n\times d}$ such that

$$\left\|\tilde{\boldsymbol{U}}_{k_0}\tilde{\boldsymbol{U}}_{k_0}^T - \boldsymbol{U}_{k_0}^\star \boldsymbol{U}_{k_0}^{\star T}\right\|_F \ge \beta - \frac{c_1'\sqrt{\sum_{i\in C_{k_0}^\star}\|\boldsymbol{e}_i\|^2}}{\sqrt{d} - \sqrt{N_{k_0}-1}}, \tag{21}$$

where $\beta := \sqrt{2\min\{d - N_{k_0}, n - d\}}$ and $c_1', c_2' > 0$ are constants.

We defer the proof to Appendix B.5. This result extends Theorem 4 from a single low-rank Gaussian component to the multi-component `MoLRG` setting. When each component has sufficiently many samples, the global optimizer recovers the true mixture assignments up to a permutation, after which the problem decomposes into K component-wise PCA problems. Conversely, Theorem 8(ii) shows that if one component has fewer than d samples, then even oracle PCA with true assignments cannot uniquely recover its subspace. Together, these results identify a sample-complexity threshold for learning the underlying subspaces, which we empirically validate in Figure 3. We now discuss the implications of our results.

- *Understanding diffusion models via subspace clustering.* To the best of our knowledge, our work is the first to establish the equivalence between training diffusion models and subspace clustering under a tractable `MoLRG`-inspired parameterization. This equivalence, together with the `MoLRG` model, allows us to show that the minimal number of samples for diffusion models to learn the underlying distribution scales linearly with the intrinsic dimension. This finding stands in sharp contrast to existing results (Oko et al., 2023; Wibisono et al., 2024) in the literature, which show that diffusion models suffer from the curse of dimensionality when learning general classes of high-dimensional distributions. Our results provide a more optimistic and practical perspective by demonstrating that diffusion models can effectively learn data distributions with intrinsic low-dimensional structures, thereby avoiding the curse of dimensionality.

- *Connections to the phase transition from memorization to generalization.* Kadkhodaie et al. (2024); Zhang et al. (2024) have empirically revealed that diffusion models learn the score function across two distinct regimes–memorization (i.e., learning the empirical distribution of the training data) and generalization (i.e., learning the underlying distribution of the data). Our work partially explains this intriguing experimental observation based on the `MoLRG` model in terms of generalization. We demonstrate that diffusion models learn the underlying data distribution, thereby enabling generalization, when the number of training samples scales linearly with the intrinsic dimension of the data distribution.[5] Our theory reveals a phase transition from failure to success in learning the underlying distribution as the number of training samples increases, shedding light on the phase transition from memorization to generalization.

  A recent work by Merger and Goldt (2025) also investigated the phase transition phenomenon by analyzing the gradient dynamics of linear models. Their findings are closely related to ours, highlighting how sample complexity governs the generalization behavior of diffusion models. However, there are key differences between our results and theirs. First, their analysis is limited to data drawn from a single Gaussian distribution, whereas our framework extends to a mixture of Gaussians. Second, their study focused on a linear neural network, whereas our model involves a mixture of two-layer neural networks, as defined in (15).

- *Future directions based on our theory.* Several promising directions for future research are based on our theoretical framework. First, our current analysis assumes that the data lies on a union of mutually orthogonal subspaces. This facilitates theoretical tractability but does not fully capture the complexity of real-world data, which often resides on overlapping or nonlinear manifolds. Extending our framework to capture these complicated structures would be a meaningful extension of our results. Second, our analysis focuses on a simplified network parameterization for learning `MoLRG`. In contrast, practical diffusion models typically rely on complex and over-parameterized architectures, such as U-Net and Transformers. A compelling direction for future research is to study the generalization behavior of diffusion models under over-parameterized nonlinear network architectures.

5. As discussed in Section 4, we consider generalization in diffusion models as their ability to accurately capture the underlying data distribution.

3.3 Empirical Validation of Theoretical Findings

Now, we conclude this section by providing phase transition experiments for $K = 1, 2, 3$, shown in Figure 2, Figure 3, and Figure 6. Our experimental results show that training diffusion models consistently exhibits a phase transition from failure to success in learning the `MoLRG` distribution (or the subspaces) as the number of training samples increases, supporting our theoretical findings in Theorems 4 and 8.

Specifically, our experimental setup is as follows. For each plot, we fix the ambient dimension of the data to be $n = 48$ and vary the subspace dimension d from 2 to 8 in increments of 1. Similarly, we vary the number of training samples N from 2 to 15 in steps of 1. For each pair of (n, d), we generate all training samples according to the `MoLRG` distribution in (9) with $\boldsymbol{e}_i = 0$ for different $K = 1, 2, 3$, independently repeating the experiment for 20 times to empirically estimate the success probability of subspace recovery. For the case $K = 1$, we apply SVD to solve the PCA problem in (12). To solve the subspace clustering problem in (17) when $K > 1$, we apply the K-subspace method with spectral initialization as described in Wang et al. (2022). To train the DAE with the theoretical parameterization (11) or (15), we optimize the training loss (5) via stochastic gradient descent (see Algorithm 1 for more details).

4. Discussion on Related Results

In this section, we discuss the relationship between our results and closely related works on diffusion models and subspace clustering.

Memorization and generalization in diffusion models. Many interesting studies have been conducted to investigate memorization and generalization of diffusion models. Zhang et al. (2024); Kadkhodaie et al. (2024) demonstrated that diffusion models tend to memorize the training data in the memorization regime and generate new samples in the generalization regime. Gu et al. (2025); Zhang et al. (2024) showed that diffusion models learn the empirical optimal score function in the memorization regime. Yoon et al. (2023) argued that diffusion models tend to generalize when they fail to memorize. Recently, Lyu et al. (2025) showed that the memorization problem can be resolved by a simple inertia update step. In the generalization regime, a popular line of research (Chen et al., 2023c,a; Bortoli, 2022; Dupuis et al., 2026; Lee et al., 2022, 2023) has established error bounds on the distance between the true data distribution and the learned data distribution under different metrics, including KL divergence and Wasserstein distance.

Diffusion models for learning low-dimensional distributions. Recently, a growing body of work has studied how diffusion models learn distributions with different low-dimensional structures. An important line of research focuses on data supported on low-dimensional subspaces. For example, a seminal work by Chen et al. (2023b) theoretically studied score approximation, estimation, and distribution recovery of diffusion models for learning from data supported on a low-dimensional linear subspace, with general latent variables beyond Gaussian distributions. Recently, Chen et al. (2025a) proposed a diffusion factor model to exploit the low-dimensional structure in asset returns and established an error bound for score estimation. In contrast to these studies, our work focuses on a union of subspaces simultaneously instead of a single subspace. In addition, while the analysis

in Chen et al. (2023b, 2025a) establishes a polynomial sample complexity bound in terms of the intrinsic dimension, our results yield a sharper bound that scales linearly with the intrinsic dimension under the `MoLRG` model. In addition, Chen (2026) assumed that the data lie in a one-dimensional linear subspace and demonstrated that the generalization ability of diffusion models stems from a smoothing-induced interpolation effect.

Another important line of research investigates more general low-dimensional manifold structures. For example, Yakovlev and Puchkin (2025) studied generalization and approximation errors of the score matching estimator under a nonparametric Gaussian mixture. Gottwald et al. (2025) studied locality structure, a form of low-dimensional structure characterized by sparse dependencies among components of the data distribution, to reduce sample complexity for training diffusion models. Cui et al. (2025) studied the training dynamics of diffusion models when the underlying distribution is an infinite Gaussian mixture supported on a latent low-dimensional manifold. A promising direction inspired by these works is to extend the `MoLRG` model to mixtures of low-dimensional manifolds and analyze the training loss of diffusion models.

Sampling rate of diffusion models with low-dimensional structures. In a complementary direction, recent works leverage low-dimensional structures in data to improve the sampling convergence analysis in diffusion models. For example, Huang et al. (2026); Li and Yan (2024); Liang et al. (2025) showed that the sampling rate of denoising diffusion probabilistic models scales with the intrinsic dimension of the data distribution. Azangulov et al. (2024); Tang and Yang (2024); Bortoli (2022) established sharp convergence rates for score-based diffusion models when the data distribution lies on or near low-dimensional manifolds.

Subspace clustering. Subspace clustering is a fundamental problem in unsupervised learning, which aims to identify and group data points that lie in a union of low-dimensional subspaces in a high-dimensional space (Agarwal and Mustafa, 2004; Vidal, 2011; Lerman and Maunu, 2018). Over the past years, a substantial body of literature has explored various approaches to the algorithmic development and theoretical analysis of subspace clustering. These include techniques such as sparse representation (Elhamifar and Vidal, 2013; Wang and Xu, 2013; Soltanolkotabi and Candes, 2012; Soltanolkotabi et al., 2014), low-rank representation (Wang et al., 2022; Liu et al., 2013, 2017; Maunu et al., 2019; Lerman et al., 2025), and spectral clustering (Li and Gu, 2021; Vidal et al., 2005). In this work, we present a new interpretation of diffusion models from the perspective of subspace clustering. This is the first time that diffusion models have been analyzed through this lens, offering new insights into how these models can effectively learn complex data distributions by leveraging the intrinsic low-dimensional subspaces within the data.

5. Practical Implications of Our Theoretical Results

Building on the results in Section 3, we study the practical value of our theoretical investigation by showing that: (i) our study of low-dimensional distribution learning offers key insights into the generalization behavior of real-world diffusion models (see Section 5.1), and (ii) the basis vectors of the identified low-dimensional subspaces correspond to different

semantic task vectors in practice, enabling controlled editing of specific attributes in content generation (see Section 5.2).

5.1 Phase Transition of Generalization in Real-World Diffusion Models

In this subsection, we conduct experiments on both synthetic `MoLRG` data and real image datasets to train U-Net-based diffusion models. Consistent with the predictions of Theorems 4 and 8, we observe a similar phase transition in generalization from failure to success, on both synthetic datasets and real image datasets. More specifically, we empirically show that the minimum number of training samples, denoted by $N_{\min}$, required for generalization scales linearly with the intrinsic dimension, denoted by ID, on both synthetic and real datasets,

$$N_{\min} \approx c \cdot \mathrm{ID}, \tag{22}$$

where $c > 0$ is a constant. As discussed at the end of Section 1.1, achieving good generalization is closely tied to accurately learning the underlying distribution in diffusion models. Therefore, our theoretical framework not only explains distribution learning in the `MoLRG` model but also offers valuable insights into the generalization of real-world diffusion models. Now, we introduce the experimental setup.

Measuring generalization in diffusion models. Recent studies (Zhang et al., 2024; Kadkhodaie et al., 2024) have shown that diffusion models trained under different settings can reproduce each other's outputs. This reproducibility provides strong evidence of generalization (Kadkhodaie et al., 2024). Furthermore, Zhang et al. (2024) empirically demonstrates that this phenomenon co-emerges with the models' ability to generate novel samples distinct from their training data. Together, these findings suggest that the ability of diffusion models to generate new samples can serve as an indicator of good generalization. Let $\{\boldsymbol{y}^{(j)}\}_{j=1}^{M}$ denote M samples generated by a diffusion model trained on the dataset $\{\boldsymbol{x}^{(i)}\}_{i=1}^{N}$. Then, we adopt a variant of the generalization (GL) score proposed in Zhang et al. (2024), defined as follows:

$$\mathrm{GL} := \frac{1}{M}\sum_{j=1}^{M} \mathbb{I}\left(\min_{i\in[N]} \left\| \boldsymbol{\Psi}\left(\boldsymbol{x}^{(i)}\right) - \boldsymbol{\Psi}\left(\boldsymbol{y}^{(j)}\right)\right\| \geq \delta\right). \tag{23}$$

Here, δ is a pre-defined threshold, $\boldsymbol{\Psi}(\boldsymbol{x})$ denotes a descriptor function applied to $\boldsymbol{x}$, and $\mathbb{I}(\cdot)$ is the indicator function, where $\mathbb{I}(x \geq \delta) = 1$ if $x \geq \delta$ and 0 otherwise. For the `MoLRG` distribution, we set $\boldsymbol{\Psi}(\boldsymbol{x})$ as the identity function and δ is defined in (56) in Appendix E.1. For real-world datasets, we set $\boldsymbol{\Psi}(\boldsymbol{x})$ as the self-supervised copy detection descriptor introduced in Pizzi et al. (2022), a neural feature extractor tailored for copy detection tasks, and set $\delta = 0.8$ according to Pizzi et al. (2022); Somepalli et al. (2023b). Additional details are provided in Appendix E.

Intuitively, the GL score measures the dissimilarity between the generated samples $\{\boldsymbol{y}^{(j)}\}_{j=1}^{M}$ and the training samples $\{\boldsymbol{x}^{(i)}\}_{i=1}^{N}$ in the feature space. A higher GL score indicates that the generated samples are less similar to the training data, reflecting better generalization. In this work, we consider a diffusion model to generalize well when $\mathrm{GL} > 0.95$, i.e., at least 95% of the generated samples are distinct from the training set.

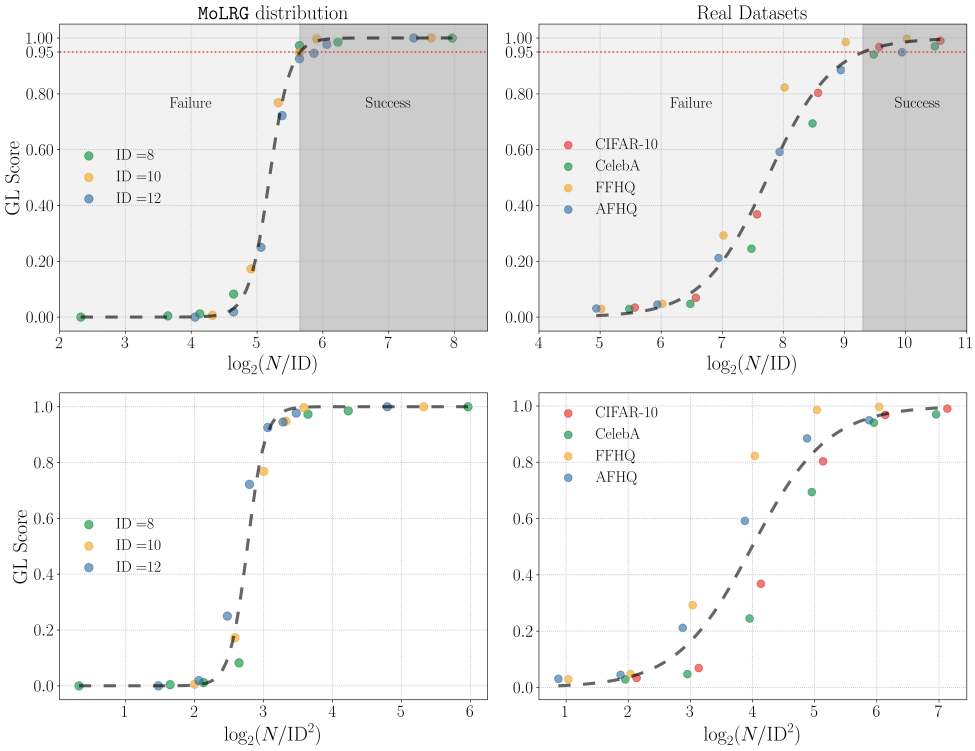


Figure 4: **Phase transition of generalization using U-Net.** Diffusion models with a U-Net architecture are trained on synthetic data sampled from the `MoLRG` distribution (left column; $K = 2$, $n = 48$, varying intrinsic dimensions) and on real image datasets: CIFAR-10, CelebA, FFHQ, and AFHQ (right column). The GL score is plotted against the ratio of training samples to the intrinsic dimension (top row) and to the square of the intrinsic dimension (bottom row). A black dashed line fits the data across different intrinsic dimensions (datasets) for each figure. A GL score above 0.95 (within the dark grey region) indicates good generalization, while a score below 0.95 (within the light grey region) indicates poor generalization.

Experiments on synthetic data. First, we demonstrate a phase transition in generalization when training U-Net on synthetic data generated from the `MoLRG` distribution. We use the `MoLRG` distribution defined in (9) and set the data dimension $n = 48$, the number of components $K = 2$, the noise level $\boldsymbol{e}_i = \mathbf{0}$, and mixing proportion $\pi_k = 1/2$. Then, we set each cluster to contain an equal number of samples and the total number is N. The intrinsic dimension of each subspace is set to d, identical across clusters. Because the subspace bases are orthogonal, the total intrinsic dimension of the distribution ID $= Kd$. We optimize the training loss in (5) using a DAE $\boldsymbol{x}_{\boldsymbol{\theta}}(\cdot, t)$, parameterized by U-Net. The same U-Net architecture is used across all experiments. The detailed experimental settings are provided in Appendix E.1.

In Figure 4 (top-left), we plot the GL scores against the ratio $\log_2(N/\mathrm{ID})$ by varying both N and ID. Here, different scatter colors correspond to different choices of ID $= 8, 10, 12$. The GL scores plotted against $\log_2(N/\mathrm{ID})$ consistently exhibit a sigmoid-shaped

	CIFAR-10	CelebA	FFHQ	AFHQ
ID	10.8	11.5	15.8	16.7

Table 2: **Intrinsic dimensions** ID **for different real-world datasets.**

curve across different ID values. This suggests that, for a fixed model architecture, the generalization ability depends primarily on the ratio N/ID rather than on the values of N or ID individually. Specifically, we fit all points using a sigmoid function (the black dashed curve shown in Figure 4 top-left), denoted by $f_{\texttt{MoLRG}}\left(N/\mathrm{ID}\right)$, with details provided in Appendix E.1. For comparison, we plot the GL scores against $\log_2\left(N/\mathrm{ID}^2\right)$ in Figure 4 (bottom-left) and fit them with a sigmoid function. The data points deviate more from the fitted curve than the curve in the top plot. The stronger alignment between the data points and the fitted curve in the top plot confirms that N/ID is a better indicator for GL score.

Recall that $\mathrm{GL} > 0.95$ indicates successful generalization. To identify $N_{\min}$, we solve $\mathrm{GL}(N_{\min}/\mathrm{ID}) \approx f_{\texttt{MoLRG}}\left(N_{\min}/\mathrm{ID}\right) = 0.95$, which implies $c = f^{-1}_{\texttt{MoLRG}}(0.95)$ in (22).[6] This demonstrates that achieving successful generalization requires $N_{\min}$ to scale linearly with ID, thereby corroborating our theoretical findings in Theorem 8.

Experiments on real image datasets. Next, our results in Figure 4 (top-right) reveal a similar phase transition in generalization across several real-world image datasets, including CIFAR-10 (Krizhevsky et al., 2009), CelebA (Liu et al., 2015), FFHQ (Kazemi and Sullivan, 2014), and AFHQ (Choi et al., 2020). Following a similar experimental setup for `MoLRG`, we use the same U-Net architecture for different datasets with extra experimental details provided in Appendix E.2. Then, we respectively plot the GL score against $\log_2(N/\mathrm{ID})$ and $\log_2(N/\mathrm{ID}^2)$ by varying the number of training samples N. However, because the intrinsic dimension of image datasets here is not known, we estimate it using the method described in Appendix E.3, with the resulting estimates summarized in Table 2.

Our results across different real image distributions also suggest that the GL score is primarily determined by the ratio N/ID. As shown in Figure 4 (top-right), the plot of the GL score against $\log_2\left(N/\mathrm{ID}\right)$ yields nearly identical sigmoid-shaped curves across different datasets. This behavior is consistent with our observations for the `MoLRG` distribution. In contrast, Figure 4 (bottom-right) shows the GL scores plotted against N/ID^2, where data points cannot be captured by a single function across datasets. This comparison further supports that N/ID is a more appropriate indicator for GL score.

Accordingly, we fit all the points of GL scores using the same function $f_{\texttt{real}}\left(N/\mathrm{ID}\right)$ (the black dashed curve shown in Figure 4 top-right), with more details provided in Appendix E.2. In this case, training diffusion models on real image datasets requires at least $N_{\min} = f^{-1}_{\texttt{real}}\left(0.95\right)\mathrm{ID}$ number of training samples to achieve good generalization. This aligns with the linear relationship in (22) with constant $c = f^{-1}_{\texttt{real}}\left(0.95\right)$.

6. It is worth noting that the linear relationship between $N_{\min}$ and ID holds as long as the data points (plotting GL score against N/ID) can be well-fitted by a function. Changing the threshold for successful generalization affects only the slope c of the linear relationship.

5.2 Correspondence between Basis Vectors of Low-Dimensional Subspaces and Semantic Attributes

In this subsection, we demonstrate that our theoretical insights provide valuable guidance for improving the controllability of image generation. We begin by outlining a method for identifying low-rank subspaces in diffusion models trained on real-world image datasets. Next, we demonstrate how to verify that the orthogonal basis vectors of the identified subspaces are semantic task vectors. As illustrated in Figure 5, these vectors can be leveraged to steer diffusion models to edit image attributes, such as gender, hairstyle, and color. While previous studies (Manor and Michaeli, 2024a,b; Tinaz et al., 2026) have explored similar methods for image editing, our study offers a new perspective for understanding the method through the lens of a low-dimensional subspace. Building on this study, our concurrent work (Chen et al., 2024) proposes a training-free method that enables controllable image editing.

Identifying low-rank subspaces in diffusion models. Although the DAE $\boldsymbol{x}_{\boldsymbol{\theta}}(\cdot, t)$ of real-world diffusion models cannot be exactly written as the theoretical parameterization introduced in (15), it is still possible to locally identify a low-rank subspace. Recent studies (Li et al., 2024c; Chen et al., 2024) have shown that $\boldsymbol{x}_{\boldsymbol{\theta}}(\cdot, t)$ can be well approximated using a first-order Taylor expansion:

$$\boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t + \boldsymbol{\delta}_t, t) \approx \boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) + \boldsymbol{J}_t \boldsymbol{\delta}_t, \tag{24}$$

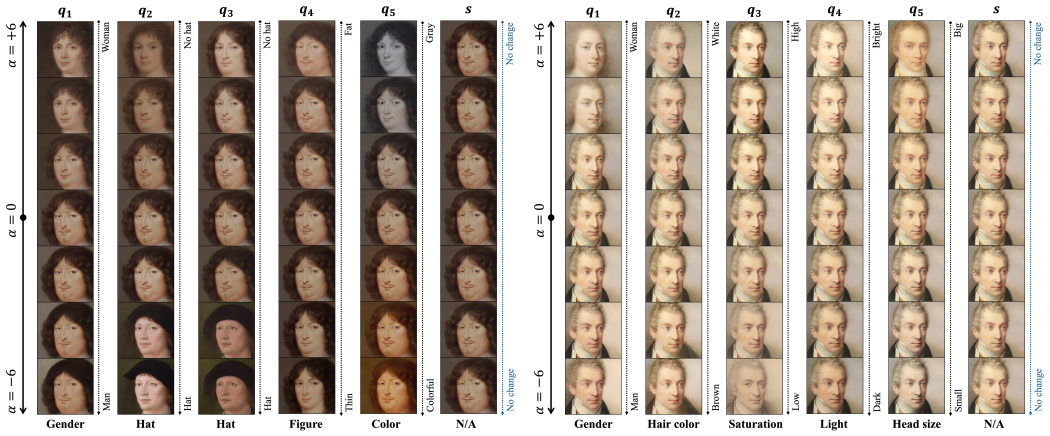
where $\boldsymbol{\delta}_t$ is the steering direction and $\boldsymbol{J}_t = \nabla_{\boldsymbol{x}_t} \boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)$ denotes the Jacobian of the DAE $\boldsymbol{x}_{\boldsymbol{\theta}}(\cdot, t)$ at $\boldsymbol{x}_t$. As we empirically verify in Appendix E.3, $\boldsymbol{J}_t$ is often a low-rank matrix at certain timesteps t, indicating that its range spans a low-dimensional subspace around $\boldsymbol{x}_t$. To identify an orthonormal basis for this subspace, we apply an SVD to $\boldsymbol{J}_t$ and obtain $\boldsymbol{J}_t = \boldsymbol{P}\boldsymbol{\Sigma}\boldsymbol{Q}^T$, where $r := \operatorname{rank}(\boldsymbol{J}_t)$, $\boldsymbol{P} = [\boldsymbol{p}_1, \cdots, \boldsymbol{p}_r] \in \mathcal{O}^{n \times r}$, $\boldsymbol{Q} = [\boldsymbol{q}_1, \cdots, \boldsymbol{q}_r] \in \mathcal{O}^{n \times r}$, and $\boldsymbol{\Sigma} = \operatorname{diag}(\sigma_1, \ldots, \sigma_r)$ with $\sigma_1 \geq \cdots \geq \sigma_r \geq 0$. Each $\boldsymbol{p}_i$ serves as an orthogonal basis vector for the subspace.

As such, if we choose $\boldsymbol{\delta}_t = \alpha \boldsymbol{q}_i$ to be one of the singular vectors $i \in [r]$, then we obtain

$$\boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t + \alpha \boldsymbol{q}_i, t) \approx \boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) + \sum_{j=1}^{r} \sigma_j \boldsymbol{p}_j \langle \boldsymbol{q}_j, \alpha \boldsymbol{q}_i \rangle = \boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) + \alpha \sigma_i \boldsymbol{p}_i.$$

Given that $\boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) \approx \mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t]$ serves as an estimate of the clean image, applying a perturbation $\boldsymbol{\delta} = \alpha \boldsymbol{q}_i$ modifies the original image $\boldsymbol{x}_0$ along the direction of the corresponding orthogonal basis vector $\boldsymbol{p}_i$ with a strength of $\alpha \sigma_i$. In the following, we empirically show that each $\boldsymbol{p}_i$ is often associated with semantic attributes.

Experimental implementation and results. We use a pre-trained diffusion denoising probabilistic model (DDPM) (Ho et al., 2020) on the MetFaces dataset (Karras et al., 2020). We randomly select an image $\boldsymbol{x}_0$ from the dataset and use DDIM (Song et al., 2020) inversion to obtain the corresponding latent $\boldsymbol{x}_t$ at $t = 0.7$. We choose the steering direction as the leading right singular vectors $\boldsymbol{q}_1, \ldots, \boldsymbol{q}_5$ and use $\tilde{\boldsymbol{x}}_t = \boldsymbol{x}_t + \alpha \boldsymbol{q}_i$ to generate new images with editing strength $\alpha \in [-6, 6]$. Figure 5 shows that these singular vectors enable different semantic edits in terms of gender, hairstyle, and color of the image. For comparison, steering the image along a direction $\boldsymbol{s}$ drawn uniformly at random from the



(a) (b)

Figure 5: **Correspondence between the singular vectors of the Jacobian of the DAE and semantic image attributes.** We use a pre-trained DDPM with U-Net on the MetFaces dataset (Karras et al., 2020). We edit the original image $\boldsymbol{x}_0$ by changing $\boldsymbol{x}_t$ into $\boldsymbol{x}_t + \alpha \boldsymbol{q}_i$, where $\boldsymbol{q}_i$ is a singular vector of the Jacobian of the DAE $\boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)$.

unit sphere results in almost *no* perceptible change in the edited images. This implies that the low-dimensional subspace spanned by $\boldsymbol{P}$ is nontrivial, with its leading basis vectors corresponding to semantic task vectors. The experimental results for more images and ablation studies for $t = 0.1$ and 0.9 are shown in Figure 8.

6. Conclusion & Future Directions

In this work, we studied the training loss of diffusion models to investigate when and why they can learn the underlying distribution without suffering from the curse of dimensionality. Assuming that the data follow a `MoLRG` distribution—an assumption supported by extensive empirical evidence—we showed that, under an appropriate network parameterization, minimizing the training loss of diffusion models is equivalent to solving a subspace clustering problem. Based on this equivalence, we further showed that the optimal solutions to the training loss can recover the underlying subspaces when the minimum number of samples scales linearly with the intrinsic dimensionality of the data distribution. Moreover, we established a correspondence between the subspace basis and the semantic attributes of image data.

Our work opens several new directions for advancing the theoretical understanding of diffusion models. First, as noted in the remarks of Theorem 8, while our work explains the generalization ability of diffusion models, it does not fully address the phenomenon of memorization or the phase transition from memorization to generalization. Future work should extend the current analysis to consider over-parameterized models and explore how these models contribute to memorization and generalization. Second, our study focuses on leveraging low-dimensional structures to understand the training process of diffusion models. However, the sampling process is also critical in diffusion models, as it influences the effi-

ciency of generated samples. Third, our current analysis relies on simplifying assumptions, including the orthogonality and symmetry of subspaces as well as the hardmax parameterization; see Assumptions 5 and 6. An important direction for future work is to relax these assumptions by considering more general subspace configurations (e.g., non-orthogonal or heterogeneous subspaces) and softmax parameterizations, and to understand how these more realistic settings affect the optimization landscape and generalization behavior of diffusion models. As discussed in Section 4, many studies have exploited low-dimensional structures to improve the sampling rate. A key direction for future research is to analyze the sampling behavior of diffusion models in the `MoLRG` model.

Acknowledgments

P. Wang is supported in part by the University of Macau under grants SRG2025-00043-FST and UMDF-TISF-I/2026/013/FST, and in part by the Macau Science and Technology Development Fund (FDCT) 0091/2025/ITP2. H. Zhang, Z. Zhang, S. Chen, and Q. Qu acknowledge support from NSF CAREER CCF-2143904, NSF CCF-2212066, NSF CCF-2212326, NSF IIS 2312842, NSF IIS 2402950, a gift grant from KLA, the MICDE Catalyst Grant, and a Google TPU Research Award. Y. Ma acknowledges support from the joint Simons Foundation-NSF DMS Grant 2031899, NSF IIS 2402951, and the ONR Grant N00014-22-1-2102. Y. Ma also acknowledges support from the startup fund from the University of Hong Kong and the JC STEM Lab fund by The Hong Kong Jockey Club Charities Trust.

The authors would also like to thank Laura Balzano (U. Michigan), Jeff Fessler (U. Michigan), Huikang Liu (SJTU), Dogyoon Song (UC Davis), Liyue Shen (U. Michigan), Rene Vidal (UPenn), and Zhihui Zhu (OSU) for stimulating discussions.

In the appendix, the organization is as follows. We first provide proof details for the results in Sections 2 and 3 in Appendices A and B, respectively. Then, we present our experimental setups for Section 2 in Appendix C, for Section 3 in Appendix D, and for Section 5 in Appendix E. Finally, additional auxiliary results for proving the main theorems are provided in Appendix F. For ease of exposition, we introduce additional notations. Given a Gaussian random vector $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, if $\boldsymbol{\Sigma} \succ \mathbf{0}$, with abuse of notation, we write its pdf as

$$\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) := \frac{1}{(2\pi)^{n/2} \det^{1/2}(\boldsymbol{\Sigma})} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right). \tag{25}$$

Appendix A. Proofs in Section 2

When the data point $\boldsymbol{x}_0$ is drawn from the `MoLRG` distribution (see Definition 1), the simplicity of Gaussian components allows us to derive a closed-form expression for the ground-truth posterior mean $\mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t]$ for all $t \in (0, 1]$ as follows. To proceed, for each $\boldsymbol{\Sigma}_k^\star$, we write its eigen-decomposition as follows:

$$\boldsymbol{\Sigma}_k^\star = \boldsymbol{U}_k^\star \boldsymbol{\Lambda}_k^\star \boldsymbol{U}_k^{\star T}, \tag{26}$$

where $\boldsymbol{\Lambda}_k^\star = \operatorname{diag}(\lambda_{k,1}^\star, \ldots, \lambda_{k,d_k}^\star)$ is a diagonal matrix with $\lambda_{k,1}^\star \geq \cdots \geq \lambda_{k,d_k}^\star > 0$ being its positive eigenvalues and $\boldsymbol{U}_k^\star \in \mathcal{O}^{n \times d_k}$ is an orthonormal matrix whose columns are the corresponding eigenvectors.

Proposition 9 *Suppose that the underlying data distribution p_{data} is a mixture of low-rank Gaussian distributions in Definition 1. In the forward process of diffusion models, the pdf of $\boldsymbol{x}_t$ for each $t > 0$ is*

$$p_t(\boldsymbol{x}) = \sum_{k=1}^K \pi_k \mathcal{N}\left(\boldsymbol{x}; s_t \boldsymbol{\mu}_k^\star, s_t^2 \boldsymbol{\Sigma}_k^\star + \gamma_t^2 \boldsymbol{I}_n\right), \tag{27}$$

where $\gamma_t := s_t \sigma_t$. Moreover, the score function of $p_t(\boldsymbol{x})$ is

$$\nabla \log p_t(\boldsymbol{x}) = \frac{1}{\gamma_t^2} \frac{\sum_{k=1}^K \pi_k \mathcal{N}\left(\boldsymbol{x}; s_t \boldsymbol{\mu}_k^\star, s_t^2 \boldsymbol{\Sigma}_k^\star + \gamma_t^2 \boldsymbol{I}_n\right) \left(\boldsymbol{I}_n - \boldsymbol{U}_k^\star \boldsymbol{D}_{k,t}^\star \boldsymbol{U}_k^{\star T}\right) (s_t \boldsymbol{\mu}_k^\star - \boldsymbol{x})}{\sum_{k=1}^K \pi_k \mathcal{N}\left(\boldsymbol{x}; s_t \boldsymbol{\mu}_k^\star, s_t^2 \boldsymbol{\Sigma}_k^\star + \gamma_t^2 \boldsymbol{I}_n\right)}, \tag{28}$$

where $\boldsymbol{D}_{k,t}^\star = \operatorname{diag}\left(\frac{s_t^2 \lambda_{k,1}^\star}{\gamma_t^2 + s_t^2 \lambda_{k,1}^\star}, \ldots, \frac{s_t^2 \lambda_{k,d_k}^\star}{\gamma_t^2 + s_t^2 \lambda_{k,d_k}^\star}\right)$.

Proof Let $Y \in \{1, \ldots, K\}$ be a discrete random variable that denotes the value of components of the mixture model. Note that $\gamma_t = s_t \sigma_t$. It follows from Definition 1 that $\mathbb{P}(Y = k) = \pi_k$ for each $k \in [K]$. Conditioned on $Y = k$, we have $\boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{\mu}_k^\star, \boldsymbol{\Sigma}_k^\star)$. This, together with (2), implies $\boldsymbol{x}_t \sim \mathcal{N}\left(s_t \boldsymbol{\mu}_k^\star, s_t^2 \boldsymbol{\Sigma}_k^\star + \gamma_t^2 \boldsymbol{I}_n\right)$. Therefore, we have

$$p_t(\boldsymbol{x}) = \sum_{k=1}^K p_t(\boldsymbol{x} \mid Y = k) \mathbb{P}(Y = k) = \sum_{k=1}^K \pi_k \mathcal{N}\left(\boldsymbol{x}; s_t \boldsymbol{\mu}_k^\star, s_t^2 \boldsymbol{\Sigma}_k^\star + \gamma_t^2 \boldsymbol{I}_n\right).$$

Next, we directly compute

$$\nabla \log p_t(\boldsymbol{x}) = \frac{\nabla p_t(\boldsymbol{x})}{p_t(\boldsymbol{x})} = \frac{\sum_{k=1}^K \pi_k \mathcal{N}\left(\boldsymbol{x}; s_t\boldsymbol{\mu}_k^\star, s_t^2\boldsymbol{\Sigma}_k^\star + \gamma_t^2\boldsymbol{I}_n\right)\left(s_t^2\boldsymbol{\Sigma}_k^\star + \gamma_t^2\boldsymbol{I}_n\right)^{-1}\left(s_t\boldsymbol{\mu}_k^\star - \boldsymbol{x}\right)}{\sum_{k=1}^K \pi_k \mathcal{N}\left(\boldsymbol{x}; s_t\boldsymbol{\mu}_k^\star, s_t^2\boldsymbol{\Sigma}_k^\star + \gamma_t^2\boldsymbol{I}_n\right)}.$$

Using (26) and the matrix inversion lemma, we compute

$$\left(s_t^2\boldsymbol{\Sigma}_k^\star + \gamma_t^2\boldsymbol{I}_n\right)^{-1} = \left(s_t^2\boldsymbol{U}_k^\star\boldsymbol{\Lambda}_k^\star\boldsymbol{U}_k^{\star T} + \gamma_t^2\boldsymbol{I}_n\right)^{-1} = \frac{1}{\gamma_t^2}\left(\boldsymbol{I}_n - \boldsymbol{U}_k^\star\boldsymbol{D}_{k,t}^\star\boldsymbol{U}_k^{\star T}\right), \tag{29}$$

where $\boldsymbol{D}_{k,t}^\star = \operatorname{diag}\left(\frac{s_t^2\lambda_{k,1}^\star}{\gamma_t^2+s_t^2\lambda_{k,1}^\star}, \ldots, \frac{s_t^2\lambda_{k,d_k}^\star}{\gamma_t^2+s_t^2\lambda_{k,d_k}^\star}\right)$. This, together with the above equation, implies (28). ■

Using the above result and (4), we compute $\mathbb{E}\left[\boldsymbol{x}_0 \mid \boldsymbol{x}_t\right]$ when $\boldsymbol{x}_0$ is drawn from the `MoLRG` distribution as follows:

Lemma 10 *Suppose $\boldsymbol{x}_0$ is drawn from the `MoLRG` distribution with parameters $\{\pi_k\}_{k=1}^K$, $\{\boldsymbol{\mu}_k^\star\}_{k=1}^K$, and $\{\boldsymbol{\Sigma}_k^\star\}_{k=1}^K$. For each time $t \in (0,1]$, it holds that*

$$\mathbb{E}\left[\boldsymbol{x}_0 \mid \boldsymbol{x}_t\right] = \sum_{k=1}^K w_{k,t}^\star(\boldsymbol{x}_t)\left(\boldsymbol{\mu}_k^\star + \boldsymbol{U}_k^\star\boldsymbol{D}_{k,t}^\star\boldsymbol{U}_k^{\star T}\left(\frac{\boldsymbol{x}_t}{s_t} - \boldsymbol{\mu}_k^\star\right)\right), \tag{30}$$

where

$$\boldsymbol{D}_{k,t}^\star = \operatorname{diag}\left(\frac{s_t^2\lambda_{k,1}^\star}{\gamma_t^2 + s_t^2\lambda_{k,1}^\star}, \ldots, \frac{s_t^2\lambda_{k,d_k}^\star}{\gamma_t^2 + s_t^2\lambda_{k,d_k}^\star}\right), \ w_{k,t}^\star(\boldsymbol{x}) := \frac{\pi_k \mathcal{N}\left(\boldsymbol{x}; s_t\boldsymbol{\mu}_k^\star, s_t^2\boldsymbol{\Sigma}_k^\star + \gamma_t^2\boldsymbol{I}_n\right)}{\sum_{l=1}^K \pi_l \mathcal{N}\left(\boldsymbol{x}; s_t\boldsymbol{\mu}_l^\star, s_t^2\boldsymbol{\Sigma}_l^\star + \gamma_t^2\boldsymbol{I}_n\right)}.$$

Proof According to (4) and (28) in Proposition 9, we compute

$$\mathbb{E}\left[\boldsymbol{x}_0 \mid \boldsymbol{x}_t\right] = \frac{\boldsymbol{x}_t + \gamma_t^2\nabla \log p_t(\boldsymbol{x}_t)}{s_t} = \sum_{k=1}^K w_{k,t}^\star(\boldsymbol{x}_t)\left(\boldsymbol{\mu}_k^\star + \boldsymbol{U}_k^\star\boldsymbol{D}_{k,t}^\star\boldsymbol{U}_k^{\star T}\left(\frac{\boldsymbol{x}_t}{s_t} - \boldsymbol{\mu}_k^\star\right)\right).$$

■

This lemma implies that the ground-truth posterior mean is a convex combination of the terms $\boldsymbol{\mu}_k^\star + \boldsymbol{U}_k^\star\boldsymbol{D}_{k,t}^\star\boldsymbol{U}_k^{\star T}\left(\boldsymbol{x}_t/s_t - \boldsymbol{\mu}_k^\star\right)$, where the weights are $w_{k,t}^\star(\boldsymbol{x}_t)$ for each $k \in [K]$.

Appendix B. Proofs in Section 3

When $\boldsymbol{\mu}_k^\star = \boldsymbol{0}$ and $\boldsymbol{\Lambda}_k^\star = \boldsymbol{I}_{d_k}$ for each $k \in [K]$, we focus on a special instance of the `MoLRG` distribution in Definition 1 as follows:

$$\boldsymbol{x}_0 \sim \sum_{k=1}^K \pi_k \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{U}_k^\star\boldsymbol{U}_k^{\star T}\right). \tag{31}$$

This, together with Lemma 10, yields that the optimal parameterization for the DAE to learn the above distribution is

$$\boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) = \frac{s_t}{s_t^2 + \gamma_t^2} \sum_{k=1}^{K} w_{k,t}(\boldsymbol{\theta}; \boldsymbol{x}_t) \boldsymbol{U}_k \boldsymbol{U}_k^T \boldsymbol{x}_t, \tag{32}$$

where $\boldsymbol{U}_k \in \mathcal{O}^{n \times d_k}$ for each $k \in [K]$ and

$$w_{k,t}(\boldsymbol{\theta}; \boldsymbol{x}_t) = \frac{\pi_k \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{0}, s_t^2 \boldsymbol{U}_k \boldsymbol{U}_k^T + \gamma_t^2 \boldsymbol{I})}{\sum_{l=1}^{K} \pi_l \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{0}, s_t^2 \boldsymbol{U}_l \boldsymbol{U}_l^T + \gamma_t^2 \boldsymbol{I})}. \tag{33}$$

B.1 Proof of Theorem 3

Proof Plugging (11) and $\boldsymbol{x}_t = s_t \boldsymbol{x}^{(i)} + \gamma_t \boldsymbol{\epsilon}$ into the integrand of (5) yields

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n)} \left[\left\| \frac{s_t}{s_t^2 + \gamma_t^2} \boldsymbol{U}\boldsymbol{U}^T \left(s_t \boldsymbol{x}^{(i)} + \gamma_t \boldsymbol{\epsilon} \right) - \boldsymbol{x}^{(i)} \right\|^2 \right] \\
= &\left\| \frac{s_t^2}{s_t^2 + \gamma_t^2} \boldsymbol{U}\boldsymbol{U}^T \boldsymbol{x}^{(i)} - \boldsymbol{x}^{(i)} \right\|^2 + \frac{(s_t \gamma_t)^2}{(s_t^2 + \gamma_t^2)^2} \mathbb{E}_{\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n)} \left[\| \boldsymbol{U}\boldsymbol{U}^T \boldsymbol{\epsilon} \|^2 \right] \\
= &\left\| \frac{s_t^2}{s_t^2 + \gamma_t^2} \boldsymbol{U}\boldsymbol{U}^T \boldsymbol{x}^{(i)} - \boldsymbol{x}^{(i)} \right\|^2 + \frac{(s_t \gamma_t)^2 d}{(s_t^2 + \gamma_t^2)^2},
\end{aligned}$$

where the first equality follows from $\mathbb{E}_{\boldsymbol{\epsilon}}[\langle \boldsymbol{x}, \boldsymbol{\epsilon} \rangle] = 0$ for any given $\boldsymbol{x} \in \mathbb{R}^n$ due to $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n)$, and the second equality uses $\mathbb{E}_{\boldsymbol{\epsilon}} \left[\| \boldsymbol{U}\boldsymbol{U}^T \boldsymbol{\epsilon} \|^2 \right] = \mathbb{E}_{\boldsymbol{\epsilon}} \left[\| \boldsymbol{U}^T \boldsymbol{\epsilon} \|^2 \right] = \sum_{i=1}^{d} \mathbb{E}_{\boldsymbol{\epsilon}} \left[\| \boldsymbol{u}_i^T \boldsymbol{\epsilon} \|^2 \right] = d$ due to $\boldsymbol{U} \in \mathcal{O}^{n \times d}$ and $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n)$. This, together with $\gamma_t = s_t \sigma_t$ and (5), yields

$$\ell(\boldsymbol{U}) = \frac{1}{N} \sum_{i=1}^{N} \int_0^1 \lambda_t \left(\| \boldsymbol{x}^{(i)} \|^2 - \frac{1 + 2\sigma_t^2}{(1 + \sigma_t^2)^2} \| \boldsymbol{U}^T \boldsymbol{x}^{(i)} \|^2 + \frac{\sigma_t^2 d}{(1 + \sigma_t^2)^2} \right) \mathrm{d}t.$$

Therefore, minimizing the above function in terms of $\boldsymbol{U}$ amounts to

$$\min_{\boldsymbol{U}^T \boldsymbol{U} = \boldsymbol{I}_d} - \int_0^1 \frac{(1 + 2\sigma_t^2) \lambda_t}{(1 + \sigma_t^2)^2} \mathrm{d}t \frac{1}{N} \sum_{i=1}^{N} \| \boldsymbol{U}^T \boldsymbol{x}^{(i)} \|^2,$$

which is equivalent to Problem (12). ■

B.2 Proof of Theorem 4

Proof For ease of exposition, let

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}^{(1)} & \dots & \boldsymbol{x}^{(N)} \end{bmatrix} \in \mathbb{R}^{n \times N}, \ \boldsymbol{A} = \begin{bmatrix} \boldsymbol{a}_1 & \dots & \boldsymbol{a}_N \end{bmatrix} \in \mathbb{R}^{d \times N}, \ \boldsymbol{E} = \begin{bmatrix} \boldsymbol{e}_1 & \dots & \boldsymbol{e}_N \end{bmatrix} \in \mathbb{R}^{n \times N}.$$

Using this and (10), we obtain

$$\boldsymbol{X} = \boldsymbol{U}^\star \boldsymbol{A} + \boldsymbol{E}. \tag{34}$$

Let $r_A := \operatorname{rank}(\boldsymbol{A}) \le \min\{d, N\}$ and $\boldsymbol{A} = \boldsymbol{U}_A \boldsymbol{\Sigma}_A \boldsymbol{V}_A^T$ be a singular value decomposition (SVD) of $\boldsymbol{A}$, where $\boldsymbol{U}_A \in \mathcal{O}^{d\times r_A}$, $\boldsymbol{V}_A \in \mathcal{O}^{N\times r_A}$, and $\boldsymbol{\Sigma}_A \in \mathbb{R}^{r_A\times r_A}$. It follows from Theorem 3 that Problem (5) with the parameterization (11) is equivalent to Problem (12).

(i) Suppose that $N \ge d$. Applying Lemma 13 with $\varepsilon = 1/(2c_1)$ to $\boldsymbol{A} \in \mathbb{R}^{d\times N}$, it holds with probability at least $1 - 1/2^{N-d+1} - \exp(-c_2 N)$ that

$$\sigma_{\min}(\boldsymbol{A}) = \sigma_d(\boldsymbol{A}) \ge \frac{\sqrt{N} - \sqrt{d-1}}{2c_1}, \tag{35}$$

where $c_1, c_2 > 0$ are constants depending polynomially only on the Gaussian moment. This implies $r_A = d$ and $\boldsymbol{U}_A \in \mathcal{O}^d$. Since Problem (12) is a PCA problem, the columns of any optimal solution $\hat{\boldsymbol{U}} \in \mathcal{O}^{n\times d}$ consist of left singular vectors associated with the top d singular values of $\boldsymbol{X}$. This, together with Wedin's Theorem (Wedin, 1972) and (34), yields

$$\left\|\hat{\boldsymbol{U}}\hat{\boldsymbol{U}}^T - \boldsymbol{U}^\star\boldsymbol{U}^{\star T}\right\|_F = \left\|\hat{\boldsymbol{U}}\hat{\boldsymbol{U}}^T - (\boldsymbol{U}^\star\boldsymbol{U}_A)(\boldsymbol{U}^\star\boldsymbol{U}_A)^T\right\|_F \le \frac{2\|\boldsymbol{E}\|_F}{\sigma_{\min}(\boldsymbol{A})} \le \frac{4c_1\|\boldsymbol{E}\|_F}{\sqrt{N}-\sqrt{d-1}}.$$

This, together with absorbing 4 into c_1, yields (13).

(ii) Suppose that $N < d$. According to Lemma 13 with $\varepsilon = 1/(2c_1)$, it holds with probability at least $1 - 1/2^{d-N+1} - \exp(-c_2 d)$ that

$$\sigma_{\min}(\boldsymbol{A}) = \sigma_N(\boldsymbol{A}) \ge \frac{\sqrt{d} - \sqrt{N-1}}{2c_1}, \tag{36}$$

where $c_1, c_2 > 0$ are constants depending polynomially only on the Gaussian moment. This implies $r_A = N$ and $\boldsymbol{U}_A \in \mathcal{O}^{d\times N}$. This, together with the fact that $\boldsymbol{A} = \boldsymbol{U}_A\boldsymbol{\Sigma}_A\boldsymbol{V}_A^T$ is an SVD of $\boldsymbol{A}$, yields that $\boldsymbol{U}^\star\boldsymbol{A} = (\boldsymbol{U}^\star\boldsymbol{U}_A)\boldsymbol{\Sigma}_A\boldsymbol{V}_A^T$ is an SVD of $\boldsymbol{U}^\star\boldsymbol{A}$ with $\boldsymbol{U}^\star\boldsymbol{U}_A \in \mathcal{O}^{n\times N}$. Note that $\operatorname{rank}(\boldsymbol{X}) \le N$. Let $\boldsymbol{U}_X \in \mathcal{O}^{n\times N}$ be an orthonormal basis containing the left singular vectors of $\boldsymbol{X}$ associated with its nonzero singular values. This, together with Wedin's Theorem (Wedin, 1972) and (36), yields

$$\left\|\boldsymbol{U}_X\boldsymbol{U}_X^T - \boldsymbol{U}^\star\boldsymbol{U}_A\boldsymbol{U}_A^T\boldsymbol{U}^{\star T}\right\|_F \le \frac{2\|\boldsymbol{E}\|_F}{\sigma_{\min}(\boldsymbol{A})} \le \frac{4c_1\|\boldsymbol{E}\|_F}{\sqrt{d}-\sqrt{N-1}}. \tag{37}$$

Let $\boldsymbol{S} = \boldsymbol{U}^\star(\boldsymbol{I} - \boldsymbol{U}_A\boldsymbol{U}_A^T)$. Then $\operatorname{rank}(\boldsymbol{S}) = d - N$, and $\operatorname{rank}\big([\boldsymbol{U}_X, \boldsymbol{S}]\big) \le d$. Since $N < d$, one can verify that for any feasible $\bar{\boldsymbol{U}}_X \in \mathcal{O}^{n\times(d-N)}$ satisfying $\boldsymbol{U}_X^T\bar{\boldsymbol{U}}_X = \boldsymbol{0}$, the matrix

$$\hat{\boldsymbol{U}} = \begin{bmatrix}\boldsymbol{U}_X & \bar{\boldsymbol{U}}_X\end{bmatrix} \in \mathcal{O}^{n\times d}$$

is an optimal solution of Problem (12).

We choose $\bar{\boldsymbol{U}}_X$ as an optimal solution of the following optimization problem:

$$\bar{\boldsymbol{U}}_X \in \arg\min_{\boldsymbol{V}\in\mathcal{O}^{n\times(d-N)},\ \boldsymbol{U}_X^T\boldsymbol{V}=\boldsymbol{0}} \|\boldsymbol{V}^T\boldsymbol{S}\|_F^2. \tag{38}$$

- If $n \ge 2d - N$, then the orthogonal complement of $\operatorname{span}([\boldsymbol{U}_X, \boldsymbol{S}])$ has dimension at least $n - d \ge d - N$. Therefore, there exists a feasible $\boldsymbol{V}$ such that $\boldsymbol{V}^T\boldsymbol{S} = \boldsymbol{0}$, and the optimal value is 0.

- If $n < 2d - N$, the orthogonal complement of $\mathrm{span}([\boldsymbol{U}_X, \boldsymbol{S}])$ has dimension at least $n - d$. Thus, we can choose $\boldsymbol{V}_1 \in \mathbb{R}^{n\times(n-d)}$ such that $\boldsymbol{V}_1^T[\boldsymbol{U}_X, \boldsymbol{S}] = \boldsymbol{0}$, and complete it with $\boldsymbol{V}_2 \in \mathbb{R}^{n\times(2d-N-n)}$ so that $\boldsymbol{V} = [\boldsymbol{V}_1, \boldsymbol{V}_2]$ is feasible. Then

$$\|\boldsymbol{V}^T\boldsymbol{S}\|_F^2 = \|\boldsymbol{V}_2^T\boldsymbol{S}\|_F^2 \le 2d - N - n.$$

These, together with $\boldsymbol{S} := \boldsymbol{U}^\star(\boldsymbol{I} - \boldsymbol{U}_A\boldsymbol{U}_A^T)$ and (38), yield

$$\left\|\bar{\boldsymbol{U}}_X^T\boldsymbol{U}^\star(\boldsymbol{I} - \boldsymbol{U}_A\boldsymbol{U}_A^T)\right\|_F^2 \le \max\{0, 2d - N - n\}. \tag{39}$$

Then, we obtain that

$$\begin{aligned}
\left\|\hat{\boldsymbol{U}}\hat{\boldsymbol{U}}^T - \boldsymbol{U}^\star\boldsymbol{U}^{\star T}\right\|_F &= \left\|\boldsymbol{U}_X\boldsymbol{U}_X^T + \bar{\boldsymbol{U}}_X\bar{\boldsymbol{U}}_X^T - \boldsymbol{U}^\star\boldsymbol{U}_A\boldsymbol{U}_A^T\boldsymbol{U}^{\star T} - \boldsymbol{U}^\star(\boldsymbol{I} - \boldsymbol{U}_A\boldsymbol{U}_A^T)\boldsymbol{U}^{\star T}\right\|_F \\
&\ge \left\|\bar{\boldsymbol{U}}_X\bar{\boldsymbol{U}}_X^T - \boldsymbol{U}^\star(\boldsymbol{I} - \boldsymbol{U}_A\boldsymbol{U}_A^T)\boldsymbol{U}^{\star T}\right\|_F - \left\|\boldsymbol{U}_X\boldsymbol{U}_X^T - \boldsymbol{U}^\star\boldsymbol{U}_A\boldsymbol{U}_A^T\boldsymbol{U}^{\star T}\right\|_F \\
&\ge \sqrt{2(d-N) - 2\max\{0, 2d - (n+N)\}} - \frac{4c_1\|\boldsymbol{E}\|_F}{\sqrt{d} - \sqrt{N-1}} \\
&= \sqrt{2\min\{d-N, n-d\}} - \frac{4c_1\|\boldsymbol{E}\|_F}{\sqrt{d} - \sqrt{N-1}},
\end{aligned}$$

where the second inequality follows from (37), (39), and the identity

$$\left\|\bar{\boldsymbol{U}}_X\bar{\boldsymbol{U}}_X^T - \boldsymbol{U}^\star(\boldsymbol{I} - \boldsymbol{U}_A\boldsymbol{U}_A^T)\boldsymbol{U}^{\star T}\right\|_F^2 = 2(d-N) - 2\left\|\bar{\boldsymbol{U}}_X^T\boldsymbol{U}^\star(\boldsymbol{I} - \boldsymbol{U}_A\boldsymbol{U}_A^T)\right\|_F^2.$$

■

B.3 Theoretical Justification of the DAE in (15)

Substituting $\pi_1 = \cdots = \pi_K$ into (33) yields

$$w_{k,t}(\boldsymbol{\theta}; \boldsymbol{x}_t) = \frac{\mathcal{N}(\boldsymbol{x}_t; \boldsymbol{0}, s_t^2\boldsymbol{U}_k\boldsymbol{U}_k^T + \gamma_t^2\boldsymbol{I})}{\sum_{l=1}^K \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{0}, s_t^2\boldsymbol{U}_l\boldsymbol{U}_l^T + \gamma_t^2\boldsymbol{I})} = \frac{\exp\left(\phi_t\left\|\boldsymbol{U}_k^T\boldsymbol{x}_t\right\|^2\right)}{\sum_{l=1}^K \exp\left(\phi_t\left\|\boldsymbol{U}_l^T\boldsymbol{x}_t\right\|^2\right)},$$

where the second equality follows from (25), (29), $d_1 = \cdots = d_K$, and $\phi_t := s_t^2/(2\gamma_t^2(s_t^2 + \gamma_t^2))$. Noting $\boldsymbol{x}_t = s_t\boldsymbol{x}_0 + \gamma_t\boldsymbol{\epsilon}$, we compute

$$\mathbb{E}_{\boldsymbol{\epsilon}}\left[\|\boldsymbol{U}_k^T(s_t\boldsymbol{x}_0 + \gamma_t\boldsymbol{\epsilon})\|^2\right] = s_t^2\|\boldsymbol{U}_k^T\boldsymbol{x}_0\|^2 + \gamma_t^2\mathbb{E}_{\boldsymbol{\epsilon}}[\|\boldsymbol{U}_k^T\boldsymbol{\epsilon}\|^2] = s_t^2\|\boldsymbol{U}_k^T\boldsymbol{x}_0\|^2 + \gamma_t^2 d,$$

where the first equality is due to $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n)$ and $\mathbb{E}_{\boldsymbol{\epsilon}}[\langle \boldsymbol{U}_k^T\boldsymbol{x}_0, \boldsymbol{U}_k^T\boldsymbol{\epsilon}\rangle] = 0$ for each $k \in [K]$. This implies that when the random fluctuation of $\|\boldsymbol{U}_k^T(s_t\boldsymbol{x}_0 + \gamma_t\boldsymbol{\epsilon})\|^2$ is small compared to its mean, we can approximate $w_{k,t}(\boldsymbol{\theta}; \boldsymbol{x}_t)$ in (32) well by

$$w_{k,t}(\boldsymbol{\theta}; \boldsymbol{x}_t) \approx \frac{\exp\left(\phi_t\left(s_t^2\|\boldsymbol{U}_k^T\boldsymbol{x}_0\|^2 + \gamma_t^2 d\right)\right)}{\sum_{l=1}^K \exp\left(\phi_t\left(s_t^2\|\boldsymbol{U}_l^T\boldsymbol{x}_0\|^2 + \gamma_t^2 d\right)\right)}.$$

This softmax function can be further approximated by the hardmax function. Therefore, we obtain the parameterization (16).

B.4 Proof of Theorem 7

Proof Plugging (15) into the integrand of (5) yields

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{\epsilon}}\left[\left\|\frac{s_t}{s_t^2+\gamma_t^2}\sum_{k=1}^{K}\hat{w}_k(\boldsymbol{\theta};\boldsymbol{x}^{(i)})\boldsymbol{U}_k\boldsymbol{U}_k^T(s_t\boldsymbol{x}^{(i)}+\gamma_t\boldsymbol{\epsilon})-\boldsymbol{x}^{(i)}\right\|^2\right]\\
=&\left\|\frac{s_t^2}{s_t^2+\gamma_t^2}\sum_{k=1}^{K}\hat{w}_k(\boldsymbol{\theta};\boldsymbol{x}^{(i)})\boldsymbol{U}_k\boldsymbol{U}_k^T\boldsymbol{x}^{(i)}-\boldsymbol{x}^{(i)}\right\|^2+\frac{(s_t\gamma_t)^2}{(s_t^2+\gamma_t^2)^2}\mathbb{E}_{\boldsymbol{\epsilon}}\left[\left\|\sum_{k=1}^{K}\hat{w}_k(\boldsymbol{\theta};\boldsymbol{x}^{(i)})\boldsymbol{U}_k\boldsymbol{U}_k^T\boldsymbol{\epsilon}\right\|^2\right]\\
=&\frac{s_t^2}{s_t^2+\gamma_t^2}\sum_{k=1}^{K}\left(\frac{s_t^2}{s_t^2+\gamma_t^2}\hat{w}_k^2(\boldsymbol{\theta};\boldsymbol{x}^{(i)})-2\hat{w}_k(\boldsymbol{\theta};\boldsymbol{x}^{(i)})\right)\|\boldsymbol{U}_k^T\boldsymbol{x}^{(i)}\|^2+\|\boldsymbol{x}^{(i)}\|^2+\frac{(s_t\gamma_t)^2d}{(s_t^2+\gamma_t^2)^2}\sum_{k=1}^{K}\hat{w}_k^2(\boldsymbol{\theta};\boldsymbol{x}^{(i)}),
\end{aligned}$$

where the first equality follows from $\mathbb{E}_{\boldsymbol{\epsilon}}[\langle\boldsymbol{x},\boldsymbol{\epsilon}\rangle]=0$ for any fixed $\boldsymbol{x}\in\mathbb{R}^n$ due to $\boldsymbol{\epsilon}\sim\mathcal{N}(\boldsymbol{0},\boldsymbol{I}_n)$, and the last equality uses $\boldsymbol{U}_k\in\mathcal{O}^{n\times d}$ and $\boldsymbol{U}_k^T\boldsymbol{U}_l=\boldsymbol{0}$ for all $k\neq l$. This, together with (5) and $\gamma_t=s_t\sigma_t$, yields

$$\begin{aligned}
\ell(\boldsymbol{\theta})=&\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}\int_0^1\frac{\lambda_t}{1+\sigma_t^2}\left(\frac{1}{1+\sigma_t^2}\hat{w}_k^2(\boldsymbol{\theta};\boldsymbol{x}^{(i)})-2\hat{w}_k(\boldsymbol{\theta};\boldsymbol{x}^{(i)})\right)\mathrm{d}t\|\boldsymbol{U}_k^T\boldsymbol{x}^{(i)}\|^2+\\
&\frac{1}{N}\int_0^1\lambda_t\mathrm{d}t\sum_{i=1}^{N}\|\boldsymbol{x}^{(i)}\|^2+\left(\int_0^1\frac{\sigma_t^2\lambda_t}{(1+\sigma_t^2)^2}\mathrm{d}t\right)\frac{d}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}\hat{w}_k^2(\boldsymbol{\theta};\boldsymbol{x}^{(i)}).
\end{aligned}$$

According to (16), we partition $[N]$ into $\{C_k(\boldsymbol{\theta})\}_{k=1}^K$ defined in (18). Then, by the definition of the hardmax weights in (16), we have

$$\hat{w}_k(\boldsymbol{\theta};\boldsymbol{x}^{(i)})=\begin{cases}1, & i\in C_k(\boldsymbol{\theta}),\\ 0, & i\notin C_k(\boldsymbol{\theta}).\end{cases}$$

Then, we obtain

$$\sum_{i=1}^{N}\sum_{k=1}^{K}\hat{w}_k^2(\boldsymbol{\theta};\boldsymbol{x}^{(i)})=\sum_{k=1}^{K}\sum_{i\in C_k(\boldsymbol{\theta})}1=N.$$

Using this partition in the above loss function, minimizing $\ell(\boldsymbol{\theta})$ is equivalent to minimizing

$$\begin{aligned}
&\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}\int_0^1\frac{\lambda_t}{1+\sigma_t^2}\left(\frac{1}{1+\sigma_t^2}\hat{w}_k^2(\boldsymbol{\theta};\boldsymbol{x}^{(i)})-2\hat{w}_k(\boldsymbol{\theta};\boldsymbol{x}^{(i)})\right)\mathrm{d}t\|\boldsymbol{U}_k^T\boldsymbol{x}^{(i)}\|^2\\
=&\left(\int_0^1\frac{\lambda_t}{1+\sigma_t^2}\left(\frac{1}{1+\sigma_t^2}-2\right)\mathrm{d}t\right)\frac{1}{N}\sum_{k=1}^{K}\sum_{i\in C_k(\boldsymbol{\theta})}\|\boldsymbol{U}_k^T\boldsymbol{x}^{(i)}\|^2.
\end{aligned}$$

Since $\frac{\lambda_t}{1+\sigma_t^2}\left(\frac{1}{1+\sigma_t^2}-2\right)<0$ for all $t\in[0,1]$, minimizing the above function is equivalent to

$$\max_{\boldsymbol{\theta}}\frac{1}{N}\sum_{k=1}^{K}\sum_{i\in C_k(\boldsymbol{\theta})}\|\boldsymbol{U}_k^T\boldsymbol{x}^{(i)}\|^2\qquad\text{s.t. }[\boldsymbol{U}_1\ \ldots\ \boldsymbol{U}_K]\in\mathcal{O}^{n\times dK}.$$

■

B.5 Proof of Theorem 8

We first establish two basic consequences of the hard assignment rule. The first shows that the ground-truth subspace bases recover the true mixture assignments. The second compares the objective value at a global optimizer with that at the ground-truth subspaces. For ease of exposition, let

$$\boldsymbol{\theta}^\star := \{\boldsymbol{U}_k^\star\}_{k=1}^K, \quad C_k^\star := \{i \in [N] : z_i = k\},$$

where z_i denotes the latent component label in Assumption 2.

Proposition 11 *Suppose that Assumptions 2, 5, and 6 hold, $d \gtrsim \log N$, and*

$$\|\boldsymbol{e}_i\| < \frac{\sqrt{d} - 2\sqrt{\log N} - 2}{2}, \quad \forall i \in [N]. \tag{40}$$

Then, with probability at least $1 - 2N^{-1}$, the following statements hold:
(i) It holds that

$$C_k(\boldsymbol{\theta}^\star) = C_k^\star, \quad \forall k \in [K]. \tag{41}$$

(ii) For any optimal solution $\hat{\boldsymbol{\theta}} := \{\hat{\boldsymbol{U}}_k\}_{k=1}^K$ of Problem (17), we have

$$\sum_{i=1}^N \|\boldsymbol{a}_i\|^2 - \sum_{l=1}^K \sum_{k=1}^K \sum_{i \in C_k(\hat{\boldsymbol{\theta}}) \cap C_l^\star} \|\hat{\boldsymbol{U}}_k^T \boldsymbol{U}_l^\star \boldsymbol{a}_i\|^2 \le 6\delta N\sqrt{d} + N\delta^2, \tag{42}$$

where $\delta := \max_{i \in [N]} \|\boldsymbol{e}_i\|$.

Proof Suppose that (63) holds for all $i \in [N]$, which happens with probability at least $1 - 2N^{-1}$ according to Lemma 15. Then, for all $i \in [N]$,

$$\sqrt{d} - (2\sqrt{\log N} + 2) \le \|\boldsymbol{a}_i\| \le \sqrt{d} + (2\sqrt{\log N} + 2). \tag{43}$$

We first prove part (i). For any $i \in C_k^\star$, Assumption 2 gives $\boldsymbol{x}^{(i)} = \boldsymbol{U}_k^\star \boldsymbol{a}_i + \boldsymbol{e}_i$. Therefore,

$$\|\boldsymbol{U}_k^{\star T} \boldsymbol{x}^{(i)}\| = \|\boldsymbol{a}_i + \boldsymbol{U}_k^{\star T} \boldsymbol{e}_i\| \ge \|\boldsymbol{a}_i\| - \|\boldsymbol{e}_i\|, \tag{44}$$

$$\|\boldsymbol{U}_l^{\star T} \boldsymbol{x}^{(i)}\| = \|\boldsymbol{U}_l^{\star T} \boldsymbol{e}_i\| \le \|\boldsymbol{e}_i\|, \quad \forall l \ne k, \tag{45}$$

where the second line uses $\boldsymbol{U}_l^{\star T} \boldsymbol{U}_k^\star = \boldsymbol{0}$ for $l \ne k$. These, together with (40) and (43), yield

$$\|\boldsymbol{U}_k^{\star T} \boldsymbol{x}^{(i)}\| > \|\boldsymbol{U}_l^{\star T} \boldsymbol{x}^{(i)}\|, \quad \forall l \ne k.$$

Hence, k is the unique maximizer in the definition of $C_k(\boldsymbol{\theta}^\star)$, and thus $i \in C_k(\boldsymbol{\theta}^\star)$. This implies $C_k^\star \subseteq C_k(\boldsymbol{\theta}^\star)$ for all $k \in [K]$. Since both $\{C_k^\star\}_{k=1}^K$ and $\{C_k(\boldsymbol{\theta}^\star)\}_{k=1}^K$ form partitions of $[N]$, we obtain (41).

We next prove part (ii). For ease of exposition, let

$$f(\boldsymbol{\theta}) := \sum_{k=1}^K \sum_{i \in C_k(\boldsymbol{\theta})} \|\boldsymbol{U}_k^T \boldsymbol{x}^{(i)}\|^2.$$

By part (i), we have $C_k(\boldsymbol{\theta}^\star) = C_k^\star$ for all $k \in [K]$. Hence,

$$\begin{aligned} f(\boldsymbol{\theta}^\star) &= \sum_{k=1}^K \sum_{i \in C_k^\star} \|\boldsymbol{U}_k^{\star T} \boldsymbol{x}^{(i)}\|^2 = \sum_{k=1}^K \sum_{i \in C_k^\star} \|\boldsymbol{a}_i + \boldsymbol{U}_k^{\star T} \boldsymbol{e}_i\|^2 \\ &= \sum_{i=1}^N \|\boldsymbol{a}_i\|^2 + 2 \sum_{k=1}^K \sum_{i \in C_k^\star} \langle \boldsymbol{a}_i, \boldsymbol{U}_k^{\star T} \boldsymbol{e}_i \rangle + \sum_{k=1}^K \sum_{i \in C_k^\star} \|\boldsymbol{U}_k^{\star T} \boldsymbol{e}_i\|^2. \end{aligned} \tag{46}$$

On the other hand,

$$\begin{aligned} f(\hat{\boldsymbol{\theta}}) &= \sum_{k=1}^K \sum_{i \in C_k(\hat{\boldsymbol{\theta}})} \|\hat{\boldsymbol{U}}_k^T \boldsymbol{x}^{(i)}\|^2 = \sum_{l=1}^K \sum_{k=1}^K \sum_{i \in C_k(\hat{\boldsymbol{\theta}}) \cap C_l^\star} \|\hat{\boldsymbol{U}}_k^T (\boldsymbol{U}_l^\star \boldsymbol{a}_i + \boldsymbol{e}_i)\|^2 \\ &= \sum_{l=1}^K \sum_{k=1}^K \sum_{i \in C_k(\hat{\boldsymbol{\theta}}) \cap C_l^\star} \left(\|\hat{\boldsymbol{U}}_k^T \boldsymbol{U}_l^\star \boldsymbol{a}_i\|^2 + 2 \langle \boldsymbol{a}_i, \boldsymbol{U}_l^{\star T} \hat{\boldsymbol{U}}_k \hat{\boldsymbol{U}}_k^T \boldsymbol{e}_i \rangle \right) \\ &\quad + \sum_{k=1}^K \sum_{i \in C_k(\hat{\boldsymbol{\theta}})} \|\hat{\boldsymbol{U}}_k^T \boldsymbol{e}_i\|^2. \end{aligned} \tag{47}$$

Since $\hat{\boldsymbol{\theta}}$ is an optimal solution of Problem (17), we have $f(\hat{\boldsymbol{\theta}}) \geq f(\boldsymbol{\theta}^\star)$. Combining this with (46) and (47) gives

$$\begin{aligned} \sum_{i=1}^N \|\boldsymbol{a}_i\|^2 - \sum_{l=1}^K \sum_{k=1}^K \sum_{i \in C_k(\hat{\boldsymbol{\theta}}) \cap C_l^\star} \|\hat{\boldsymbol{U}}_k^T \boldsymbol{U}_l^\star \boldsymbol{a}_i\|^2 &\leq 2 \sum_{l=1}^K \sum_{k=1}^K \sum_{i \in C_k(\hat{\boldsymbol{\theta}}) \cap C_l^\star} \left| \langle \boldsymbol{a}_i, \boldsymbol{U}_l^{\star T} \hat{\boldsymbol{U}}_k \hat{\boldsymbol{U}}_k^T \boldsymbol{e}_i \rangle \right| \\ &\quad + \sum_{k=1}^K \sum_{i \in C_k(\hat{\boldsymbol{\theta}})} \|\hat{\boldsymbol{U}}_k^T \boldsymbol{e}_i\|^2 + 2 \sum_{k=1}^K \sum_{i \in C_k^\star} \left| \langle \boldsymbol{a}_i, \boldsymbol{U}_k^{\star T} \boldsymbol{e}_i \rangle \right|. \end{aligned}$$

Using $\|\boldsymbol{e}_i\| \leq \delta$ and $\boldsymbol{U}_k^\star, \hat{\boldsymbol{U}}_k \in \mathcal{O}^{n \times d}$, we obtain

$$\sum_{i=1}^N \|\boldsymbol{a}_i\|^2 - \sum_{l=1}^K \sum_{k=1}^K \sum_{i \in C_k(\hat{\boldsymbol{\theta}}) \cap C_l^\star} \|\hat{\boldsymbol{U}}_k^T \boldsymbol{U}_l^\star \boldsymbol{a}_i\|^2 \leq 4\delta \sum_{i=1}^N \|\boldsymbol{a}_i\| + N\delta^2.$$

Finally, by (43) and $d \gtrsim \log N$, we have $\|\boldsymbol{a}_i\| \leq \sqrt{d} + 2\sqrt{\log N} + 2 \leq 3\sqrt{d}/2$, and hence

$$4\delta \sum_{i=1}^N \|\boldsymbol{a}_i\| + N\delta^2 \leq 6\delta N \sqrt{d} + N\delta^2.$$

This proves (42). ■

We now show that, under a sufficiently small noise level, any global solution of the subspace clustering problem recovers the true assignments up to a permutation. The key challenge is that the overlaps $C_r(\hat{\boldsymbol{\theta}}) \cap C_k^\star$ are data-dependent subsets. To address this, we use Lemma 16, which holds uniformly over arbitrary subsets.

Proposition 12 *Suppose that Assumptions 2, 5, and 6 hold. Let* $\hat{\boldsymbol{\theta}} = \{\hat{\boldsymbol{U}}_k\}_{k=1}^K$ *be an optimal solution of Problem* (17) *and* $N_{\min} := \min_{k\in[K]} N_k$*. Suppose that* $d \gtrsim \log N$ *and*

$$\|\boldsymbol{e}_i\| \le c_\delta \min\left\{\frac{N_{\min}}{K^3 N\sqrt{d}}, \frac{\sqrt{d}}{N}\right\}, \quad \forall i \in [N], \tag{48}$$

where $c_\delta > 0$ *is a sufficiently small absolute constant. Then, with probability at least*

$$1 - 2N^{-1} - CK\exp\left(Cd\log(CK) - c\frac{N_{\min}}{K^6}\right),$$

there exists a permutation $\Pi : [K] \to [K]$ *such that*

$$C_{\Pi(k)}(\hat{\boldsymbol{\theta}}) = C_k^\star, \qquad \forall k \in [K],$$

where $c, C > 0$ *are absolute constants.*

Proof Recall that $\delta := \max_{i\in[N]} \|\boldsymbol{e}_i\|$. By (48), $d \gtrsim \log N$, and a sufficiently small choice of $c_\delta > 0$, the noise condition in Proposition 11 holds. Therefore, according to Proposition 11, it holds with probability at least $1 - 2N^{-1}$ that

$$\sum_{i=1}^N \|\boldsymbol{a}_i\|^2 - \sum_{l=1}^K \sum_{r=1}^K \sum_{i\in C_r(\hat{\boldsymbol{\theta}})\cap C_l^\star} \|\hat{\boldsymbol{U}}_r^T \boldsymbol{U}_l^\star \boldsymbol{a}_i\|^2 \le B := 6\delta N\sqrt{d} + N\delta^2. \tag{49}$$

Using (48), we obtain for sufficiently small $c_\delta > 0$ that

$$B \le c_0 \frac{N_{\min}}{K^3}, \quad 8B < d, \tag{50}$$

where $c_0 > 0$ is a sufficiently small absolute constant. For ease of exposition, let

$$N_{rk} := |C_r(\hat{\boldsymbol{\theta}}) \cap C_k^\star|, \quad \forall r, k \in [K].$$

For each $k \in [K]$, choose $\Pi(k) \in \arg\max_{r\in[K]} N_{rk}$. Since $\sum_{r=1}^K N_{rk} = N_k$, we have

$$N_{\Pi(k)k} \ge \frac{N_k}{K}. \tag{51}$$

Fix any $k \in [K]$. From (49), and using $\|\hat{\boldsymbol{U}}_r^T \boldsymbol{U}_l^\star \boldsymbol{a}_i\| \le \|\boldsymbol{a}_i\|$, we obtain

$$\begin{aligned} B &\ge \sum_{i\in C_{\Pi(k)}(\hat{\boldsymbol{\theta}})\cap C_k^\star} \left(\|\boldsymbol{a}_i\|^2 - \|\hat{\boldsymbol{U}}_{\Pi(k)}^T \boldsymbol{U}_k^\star \boldsymbol{a}_i\|^2\right) \\ &= \left\langle \boldsymbol{I} - \boldsymbol{U}_k^{\star T} \hat{\boldsymbol{U}}_{\Pi(k)} \hat{\boldsymbol{U}}_{\Pi(k)}^T \boldsymbol{U}_k^\star, \sum_{i\in C_{\Pi(k)}(\hat{\boldsymbol{\theta}})\cap C_k^\star} \boldsymbol{a}_i \boldsymbol{a}_i^T \right\rangle. \end{aligned} \tag{52}$$

Conditioning on the latent labels, the vectors $\{\boldsymbol{a}_i : i \in C_k^\star\}$ are i.i.d. standard Gaussian vectors. We apply Lemma 16 with $\eta = 1/4$, $\kappa = 1/K$, $M = N_k$, and $m = d$. In this case,

$$\gamma_K = \Phi^{-1}\left(\frac{1}{2} + \frac{3}{8K}\right),$$

and a Taylor expansion of Φ around zero gives

$$\gamma_K \asymp K^{-1}, \quad \mu_K := \frac{3}{4K} - \sqrt{\frac{2\gamma_K^2}{\pi}} \exp\left(-\frac{\gamma_K^2}{2}\right) \asymp K^{-3}.$$

Therefore, with probability at least $1 - C_1 \exp\left(C_1 d \log(C_1 K) - c_1 N_k / K^6\right)$, the following bound holds simultaneously for every subset $\Omega \subseteq C_k^\star$ satisfying $|\Omega| \geq N_k/K$:

$$\lambda_{\min}\left(\sum_{i\in\Omega} \boldsymbol{a}_i \boldsymbol{a}_i^T\right) \geq c_2 \frac{N_k}{K^3},$$

where $c_1, c_2, C_1 > 0$ are absolute constants. By (51), $C_{\Pi(k)}(\hat{\boldsymbol{\theta}}) \cap C_k^\star$ is such a subset. Hence,

$$\lambda_{\min}\left(\sum_{i\in C_{\Pi(k)}(\hat{\boldsymbol{\theta}})\cap C_k^\star} \boldsymbol{a}_i \boldsymbol{a}_i^T\right) \geq c_2 \frac{N_k}{K^3}.$$

This, together with Lemma 17 and (52), yields

$$\mathrm{Tr}\left(\boldsymbol{I} - \boldsymbol{U}_k^{\star T} \hat{\boldsymbol{U}}_{\Pi(k)} \hat{\boldsymbol{U}}_{\Pi(k)}^T \boldsymbol{U}_k^\star\right) \leq \frac{K^3 B}{c_2 N_k}.$$

Since $[\boldsymbol{U}_1^\star, \ldots, \boldsymbol{U}_K^\star] \in \mathcal{O}^{n\times dK}$, it follows that

$$\sum_{l\neq k} \|\hat{\boldsymbol{U}}_{\Pi(k)}^T \boldsymbol{U}_l^\star\|_F^2 \leq \mathrm{Tr}\left(\boldsymbol{I} - \hat{\boldsymbol{U}}_{\Pi(k)}^T \boldsymbol{U}_k^\star \boldsymbol{U}_k^{\star T} \hat{\boldsymbol{U}}_{\Pi(k)}\right) \leq \frac{K^3 B}{c_2 N_k} \leq \frac{1}{2}, \tag{53}$$

where the last inequality follows from (50) by taking $c_0 > 0$ sufficiently small.

Next, again by (49),

$$B \geq \sum_{l\neq k} \sum_{i\in C_{\Pi(k)}(\hat{\boldsymbol{\theta}})\cap C_l^\star} \left(\|\boldsymbol{a}_i\|^2 - \|\hat{\boldsymbol{U}}_{\Pi(k)}^T \boldsymbol{U}_l^\star \boldsymbol{a}_i\|^2\right).$$

For each $l \neq k$, (53) gives $\|\hat{\boldsymbol{U}}_{\Pi(k)}^T \boldsymbol{U}_l^\star\|^2 \leq \|\hat{\boldsymbol{U}}_{\Pi(k)}^T \boldsymbol{U}_l^\star\|_F^2 \leq 1/2$. Thus, for every $i \in C_{\Pi(k)}(\hat{\boldsymbol{\theta}}) \cap C_l^\star$ with $l \neq k$,

$$\|\boldsymbol{a}_i\|^2 - \|\hat{\boldsymbol{U}}_{\Pi(k)}^T \boldsymbol{U}_l^\star \boldsymbol{a}_i\|^2 \geq \frac{1}{2}\|\boldsymbol{a}_i\|^2 \geq \frac{1}{2}(\sqrt{d} - 2\sqrt{\log N} - 2)^2 \geq \frac{d}{8},$$

where the last inequality follows from $d \gtrsim \log N$. Hence,

$$B \geq \frac{d}{8} \sum_{l\neq k} N_{\Pi(k)l}.$$

Using $8B < d$, we obtain $\sum_{l\neq k} N_{\Pi(k)l} < 1$. Since the left-hand side is an integer, $N_{\Pi(k)l} = 0$ for all $l \neq k$. Therefore,

$$C_{\Pi(k)}(\hat{\boldsymbol{\theta}}) \subseteq C_k^\star, \quad \forall k \in [K].$$

It remains to show that Π is a permutation. If $\Pi(k) = \Pi(l)$ for some $k \neq l$, then

$$C_{\Pi(k)}(\hat{\boldsymbol{\theta}}) = C_{\Pi(l)}(\hat{\boldsymbol{\theta}}) \subseteq C_k^\star \cap C_l^\star = \varnothing,$$

which contradicts $N_{\Pi(k)k} \geq N_k/K > 0$. Hence Π is injective and therefore a permutation of $[K]$. Since both $\{C_{\Pi(k)}(\hat{\boldsymbol{\theta}})\}_{k=1}^K$ and $\{C_k^\star\}_{k=1}^K$ are partitions of $[N]$, we conclude that

$$C_{\Pi(k)}(\hat{\boldsymbol{\theta}}) = C_k^\star, \qquad \forall k \in [K].$$

Finally, applying a union bound over $k \in [K]$ for the covariance events and combining it with the event in Proposition 11 gives probability at least

$$1 - 2N^{-1} - C_1 K \exp\left(C_1 d \log(C_1 K) - c_1 \frac{N_{\min}}{K^6}\right).$$

This proves the desired result. ■

Proof of Theorem 8. By Theorem 7, Problem (5) is equivalent to the subspace clustering problem in (17). Hence, for part (i), it suffices to analyze the optimal solutions of Problem (17).

We first prove part (i). Let $\mathcal{E}$ denote the event in Proposition 12. Under the assumptions in part (i), Proposition 12 implies that $\mathcal{E}$ holds with probability at least $1 - N^{-\Omega(1)}$. On this event, for any optimal solution $\hat{\boldsymbol{\theta}} = \{\hat{\boldsymbol{U}}_k\}_{k=1}^K$ of Problem (17), there exists a permutation $\Pi : [K] \to [K]$ such that

$$C_{\Pi(k)}(\hat{\boldsymbol{\theta}}) = C_k^\star, \quad \forall k \in [K]. \tag{54}$$

After relabeling the estimated subspaces according to Π, the objective over the recovered clusters decomposes into the component-wise PCA problems

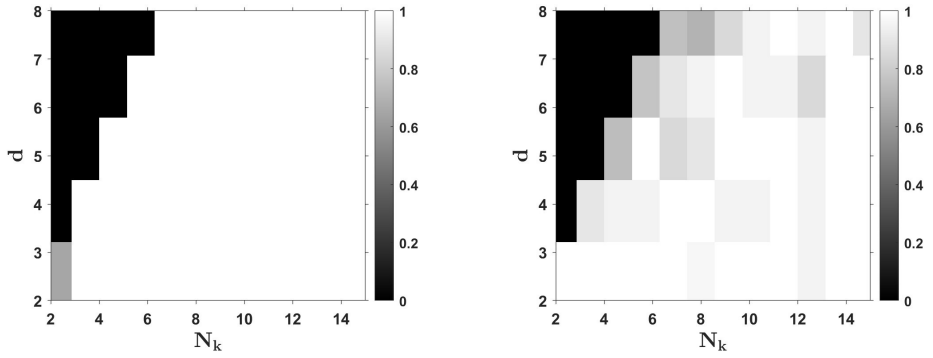
$$\max_{\boldsymbol{U} \in \mathcal{O}^{n \times d}} \frac{1}{N_k} \sum_{i \in C_k^\star} \|\boldsymbol{U}^T \boldsymbol{x}^{(i)}\|^2, \qquad k \in [K]. \tag{55}$$

For each $k \in [K]$, the samples in $C_k^\star$ satisfy $\boldsymbol{x}^{(i)} = \boldsymbol{U}_k^\star \boldsymbol{a}_i + \boldsymbol{e}_i$ for all $i \in C_k^\star$, where $\boldsymbol{a}_i \overset{i.i.d.}{\sim} \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_d)$. Since the additional sample-size condition in part (i) implies $N_k \geq d + \Omega(\log N)$ for all $k \in [K]$, applying Theorem 4 to (55) yields that, with probability at least $1 - N^{-\Omega(1)}$, for all $k \in [K]$,

$$\left\| \hat{\boldsymbol{U}}_{\Pi(k)} \hat{\boldsymbol{U}}_{\Pi(k)}^T - \boldsymbol{U}_k^\star \boldsymbol{U}_k^{\star T} \right\|_F \leq \frac{c_1 \sqrt{\sum_{i \in C_k^\star} \|\boldsymbol{e}_i\|^2}}{\sqrt{N_k} - \sqrt{d-1}}.$$

Combining this event with $\mathcal{E}$ proves part (i).

We next prove part (ii). This part follows directly from Theorem 4(ii) applied to the PCA problem over $C_{k_0}^\star$, since for $i \in C_{k_0}^\star$, Assumption 2 gives $\boldsymbol{x}^{(i)} = \boldsymbol{U}_{k_0}^\star \boldsymbol{a}_i + \boldsymbol{e}_i$ for all $\boldsymbol{a}_i \overset{i.i.d.}{\sim} \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_d)$. This proves part (ii). ■



(a) **PCA** (b) **Diffusion model**

Figure 6: **Phase transition of learning the `MoLRG` distribution when $K = 3$.** The x-axis is the number of training samples and y-axis is the dimension of subspaces. We apply a subspace clustering method and train diffusion models for solving Problems (17) and (5), visualizing the results in (a) and (b), respectively.

Appendix C. Experimental Setups in Section 2

In this section, we provide the detailed experimental setup for Section 2.4. Given a real-world dataset $\{\boldsymbol{x}^{(i)}\}_{i=1}^N$ with K classes, we outline the procedure to estimate a `MoLRG` distribution from the data. First, we set $\pi_k = 1/K$ and compute $\boldsymbol{\mu}_k$ as the mean of all images in class k. We then estimate the $\boldsymbol{U}_k$ and $\boldsymbol{\Lambda}_k$ by computing a rank-d_k truncated SVD of the covariance matrix for the samples in class k. We plug these parameters into $\mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t]$ in (7) and compute the score function $\nabla \log p_t(\boldsymbol{x}_t)$ using (4). Finally, we use the estimated score function to generate images by numerically solving (3).

For the CIFAR-10 Low-Frequency and FFHQ Low-Frequency datasets, we apply Gaussian blurring with a kernel size of 13 and a variance of 2.0 to extract the low-frequency components from CIFAR-10 and FFHQ.

We set $K = 10$, $d_k = 20$ for MNIST and FashionMNIST, $K = 10$, $d_k = 200$ for CIFAR-10, and $K = 5$, $d_k = 500$ for FFHQ. Since FFHQ lacks annotated labels, we apply the expectation-maximization algorithm for clustering and label generation. For comparison, we use a Gaussian distribution with its mean and covariance set to the mean and covariance of all training samples. In addition, we train an EDM-based diffusion model (Karras et al., 2022) on each dataset as a comparison. We employ the second-order Heun Solver (Karras et al., 2022) with 35 steps as the diffusion sampler to numerically solve (3) and generate samples from learned distributions. For qualitative evaluation, we visualize samples from 12 initial noise inputs per dataset for both theoretical (`MoLRG` and Gaussian) and real diffusion models. For quantitative evaluation, we generate 10K noise samples and compute the Euclidean distance between the theoretical and real model outputs (defined in (8)).

Algorithm 1 SGD for optimizing the training loss (5)

Input: Training samples $\{\boldsymbol{x}^{(i)}\}_{i=1}^N$
for $j = 0, 1, 2, \ldots, J$ **do**
1. Randomly select $\{(i_m, t_m)\}_{m=1}^M$, where $i_m \in [N]$ and $t_m \in (0, 1)$ and a noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$
2. Take a gradient step

$$\boldsymbol{\theta}^{j+1} \leftarrow \boldsymbol{\theta}^j - \frac{\eta}{M} \sum_{m\in[M]} \nabla_{\boldsymbol{\theta}} \left\| \boldsymbol{x}_{\boldsymbol{\theta}^j}(s_{t_m}\boldsymbol{x}^{(i_m)} + \gamma_{t_m}\boldsymbol{\epsilon}, t_m) - \boldsymbol{x}^{(i_m)} \right\|^2$$

end for

Appendix D. Experimental Setups in Section 3

In this section, we provide detailed setups for the experiment in Section 3.3. This experiment aims to validate Theorems 4 and 8. Here, we present the stochastic gradient descent (SGD) algorithm for solving Problem (5) in Algorithm 1.

Now, we specify how to choose the parameters of the SGD in our implementation. We divide the time interval $[0, 1]$ into 64 time steps. When $K = 1$, we set the learning rate $\eta = 10^{-4}$, batch size $M = 128N_k$, and number of iterations $J = 10^4$. When $K = 2$, we set the learning rate $\eta = 2\times10^{-5}$, batch size $M = 1024$, number of iterations $J = 10^5$. In particular, when $K = 2$, we use the inner-product based spectral method (TIPS) initialization (Wang et al., 2022) to improve the convergence of SGD. Specifically, TIPS generates the initial clusters based on the pairwise similarities of $\{\boldsymbol{x}^{(i)}\}_{i=1}^N$ and then estimates the initialization $\boldsymbol{\theta}^0 = \{\boldsymbol{U}_k^0\}$ according to the clustering results. Further details can be found in Section 2.3 of (Wang et al., 2022). We calculate the success rate as follows. If the returned subspace basis matrices $\{\boldsymbol{U}_k\}_{k=1}^K$ satisfy

$$\frac{1}{K}\sum_{k=1}^K \left\| \boldsymbol{U}_{\Pi(k)}\boldsymbol{U}_{\Pi(k)}^T - \boldsymbol{U}_k^\star \boldsymbol{U}_k^{\star T} \right\| \le 0.5$$

for some permutation $\Pi : [K] \to [K]$, it is considered successful.

Appendix E. Experimental Setups in Section 5

In this section, we provide detailed setups for the experiments in Section 5. Specifically, we describe the settings for using a U-Net-based diffusion model to (1) learn `MoLRG` distribution (Appendix E.1), (2) learn real-world image distribution (Appendix E.2), and (3) estimate the intrinsic dimension of real-world image distribution (Appendix E.3).

E.1 Learning the `MoLRG` distribution with U-Net

In our implementation, we set ID $\in \{8, 10, 12\}$.

- When ID $= 8$, $N_k \in \{20, 50, 70, 100, 200, 300, 1000\}$;
- When ID $= 10$, $N_k \in \{100, 150, 200, 250, 300, 1000\}$;

- When ID $= 12$, $N_k \in \{100, 150, 200, 250, 300, 350, 400, 1000\}$.

To train U-Net, we use the stochastic gradient descent in Algorithm 1. We use DDPM++ architecture (Song et al., 2021a) for the U-Net and EDM (Karras et al., 2022) noise scheduler. We set the learning rate 10^{-3}, batch size 64, and number of iterations $J = 10^4$.

For a specific `MoLRG` distribution p_{data} with N pre-selected training samples $\boldsymbol{x}^{(i)} \sim p_{\text{data}}$, the threshold δ is chosen such that the following inequality holds:

$$\frac{1}{M_z}\sum_{j=1}^{M_z}\mathbb{I}\left(\min_{i\in[N]}\|\boldsymbol{\Psi}\left(\boldsymbol{x}^{(i)}\right)-\boldsymbol{\Psi}\left(\boldsymbol{z}^{(j)}\right)\|\geq\delta\right)=0.95. \tag{56}$$

Intuitively, this definition ensures that with 95% probability, a newly drawn sample $\boldsymbol{z}^{(j)} \sim p_{\text{data}}$ will be at least δ away (in the $\boldsymbol{\Psi}$-transformation space) from its nearest neighbor among the training samples $\boldsymbol{x}^{(i)}$. While (56) has a theoretical analytical solution, we approximate δ numerically in practice. Specifically, we set $M_z = 10^3$, compute the minimum distance $\min_{i\in[N]}\|\boldsymbol{\Psi}\left(\boldsymbol{x}^{(i)}\right)-\boldsymbol{\Psi}\left(\boldsymbol{z}^{(j)}\right)\|$ for each j, and set δ as the 5th percentile (i.e., the 0.05-quantile) of the resulting distance distribution. To empirically estimate (23), we set $M = 10^3$.

To quantitatively estimate the transition in Figure 4 (top-left), we fit the curve using the following sigmoid-parameterized function:

$$\mathrm{GL}\left(\frac{N}{\mathrm{ID}}\right)\approx f_{\texttt{MoLRG}}\left(\frac{N}{\mathrm{ID}}\right)=\frac{1}{1+\exp\left(-a(\log_2\left(N/\mathrm{ID}\right)-b)\right)}, \tag{57}$$

where the fitted parameters are $a = 6.22$ and $b = 5.20$. Solving this equation numerically gives $f^{-1}_{\texttt{MoLRG}}(0.95) = 50.2$, indicating that U-Net architectures trained on the `MoLRG` distribution generalize when $N_k \geq 50.2d_k$. We use the same parameterized function (57) for the fitted curve in Figure 4 (bottom-left), by changing the input variable to N/ID^2.

E.2 Learning real-world image data distributions with U-Net

To train diffusion models for real-world image datasets, we use the DDPM++ architecture (Song et al., 2021a) for U-Net and variance preserving (VP) (Song et al., 2021a) noise scheduler. The U-Net is trained using the Adam optimizer (Kingma and Ba, 2015), a variant of SGD in Algorithm 1. We set the learning rate $\eta = 10^{-3}$, batch size $M = 512$, and the total number of iterations 10^5. To empirically estimate (23), we set $M = 10^4$.

The curve $f_{\texttt{real}}$ is parameterized the same way as $f_{\texttt{MoLRG}}$ in (57), with $a = 1.88$ and $b = 7.74$. Then, we numerically solve for $f^{-1}_{\texttt{real}}(0.95) = 630.3$, indicating that U-Net architectures trained on real data distribution generalize when $N \geq 630.3\mathrm{ID}$. We use the same parameterized function (57) for the fitted curve in Figure 4 (bottom-right), by changing the input variable to N/ID^2.

E.3 Estimating the intrinsic dimension of real-world dataset

In this subsection, we conduct numerical experiments to estimate the intrinsic dimension of real-world image data distribution. By Lemma 10, as $t \to 1$, we have

$$\nabla_{\boldsymbol{x}_t} \mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t] \approx \frac{1}{s_t} \sum_{k=1}^{K} \boldsymbol{U}_k^\star \boldsymbol{D}_{k,t}^\star \boldsymbol{U}_k^{\star T}, \tag{58}$$

given $w_{k,t}^*(\boldsymbol{x}_t) \approx 1$ and $\nabla_{\boldsymbol{x}_t} w_{k,t}^*(\boldsymbol{x}_t) \approx \boldsymbol{0}$. This relationship allows us to estimate the intrinsic dimension ID of a `MoLRG` distribution as:

$$\mathrm{ID} := \operatorname{rank}\left(\sum_{k=1}^{K} \boldsymbol{U}_k^\star \boldsymbol{D}_{k,t}^\star \boldsymbol{U}_k^{\star T} \right) \approx \operatorname{rank}\left(\nabla_{\boldsymbol{x}_t} \mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t] \right), \tag{59}$$

as $t \to 1$. Note that the DAE $\boldsymbol{x}_{\boldsymbol{\theta}}(\cdot, t)$ of the trained diffusion models satisfies $\boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) \approx \mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t]$. Combining this with the observation in Section 2.4 that real-world image distributions can be well approximated by `MoLRG` distributions, we conclude that the intrinsic dimension can be estimated by:

$$\mathrm{ID} \approx \operatorname{rank}\left(\nabla_{\boldsymbol{x}_t} \boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) \right). \tag{60}$$

We evaluate the intrinsic dimension ID over four different datasets: CIFAR-10, CelebA, FFHQ, and AFHQ. We resize images from FFHQ and AFHQ such that $n = 3072$ for all datasets. We calculate the numerical rank of the Jacobian $\nabla_{\boldsymbol{x}_t} \boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)$ through

$$\operatorname{rank}\left(\nabla_{\boldsymbol{x}_t} \boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) \right) := \arg\min \left\{ r \in [1, n] : \frac{\sum_{i=1}^{r} \sigma_i^2 \left(\nabla_{\boldsymbol{x}_t} \boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) \right)}{\sum_{i=1}^{n} \sigma_i^2 \left(\nabla_{\boldsymbol{x}_t} \boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) \right)} > \eta^2 \right\}, \tag{61}$$

with $\eta = 0.99$ and recall $\sigma_i(\boldsymbol{A})$ denotes the i-th singular value of matrix $\boldsymbol{A}$.

To select a timestep t to estimate the intrinsic dimension, we evaluate $\operatorname{rank}\left(\nabla_{\boldsymbol{x}_t} \boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) \right)$ at different timesteps across multiple datasets, as shown in Figure 7. Specifically, given a random initial noise $\boldsymbol{x}_1 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n)$, we use the diffusion model to generate a sequence of images $\{\boldsymbol{x}_t\}$ according to the reverse ODE in (3). Along the sampling trajectory $\{\boldsymbol{x}_t\}$, we estimate $\operatorname{rank}\left(\nabla_{\boldsymbol{x}_t} \boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) \right)$ at each timestep. For the experiments, we utilize the Elucidating Diffusion Model (EDM) with the EDM noise scheduler (Karras et al., 2022) and DDPM++ architecture (Song et al., 2020). Moreover, we employ an 18-step Heun's solver for sampling and present the results for 12 of these steps ($t = 0, 0.001, 0.003, 0.027, 0.171, 0.277, 0.494, 0.650, 0.712, 0.815, 0.858, 0.934$). For each dataset, we randomly sample 15 initial noise $\boldsymbol{x}_1$, calculate the mean of $\operatorname{rank}(\nabla_{\boldsymbol{x}_t} \boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t))$ along the trajectory $\{\boldsymbol{x}_t\}$.

As shown in Figure 7, the plot of rank against t exhibits a U-shaped curve, with the lowest rank consistently occurring around $t = 0.815$ across all datasets. Timesteps too close to $t = 0$ or $t = 1$ are unsuitable for estimating the intrinsic dimension: when t approaches 1, $\boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)$ becomes less accurate because the training loss (5) assigns a small weight λ_t to such timesteps; when t approaches 0, Karras et al. (2022) parameterize $\boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)$ to be $\boldsymbol{x}_t$, causing $\operatorname{rank}\left(\nabla_{\boldsymbol{x}_t} \boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) \right) = n$ to be naturally very high. Thus, we select $t = 0.815$, the timestep that achieves the lowest rank, to estimate the intrinsic dimension of real-world datasets. The estimated ID is shown in Table 2.

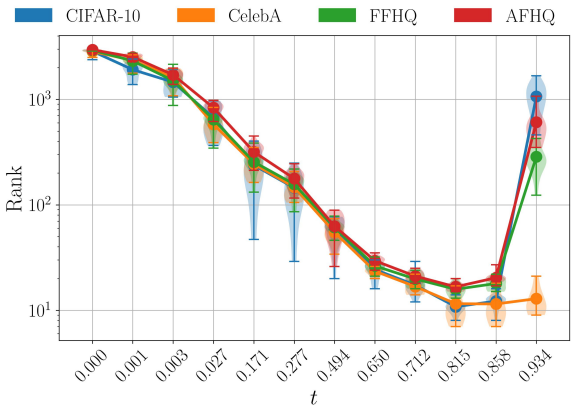


Figure 7: **Low-rank property of the denoising autoencoder of trained diffusion models.** We plot the numerical rank of the Jacobian of the denoising autoencoder, i.e., $\nabla_{\boldsymbol{x}_t} \boldsymbol{x}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)$, against the timestep t by training diffusion models on different datasets. We train diffusion models on image datasets CIFAR-10, CelebA, FFHQ, and AFHQ. The experimental details are provided in Appendix E.3.

Appendix F. Auxiliary Results

First, we present a probabilistic result to prove Theorem 4, which provides an optimal estimate of the small singular values of a matrix with i.i.d. Gaussian entries. This lemma is proved in (Rudelson and Vershynin, 2009, Theorem 1.1) for subgaussian random variables. Note that a random variable ξ is called subgaussian if there exists $c > 0$ such that for all $t > 0$,

$$\mathbb{P}\left(|\xi| > t\right) \leq 2\exp\left(-\frac{t^2}{c^2}\right).$$

We say that the minimal c in this inequality is the subgaussian moment of ξ.

Lemma 13 *Let $\boldsymbol{A}$ be an $m \times n$ random matrix, where $m \geq n$, whose elements are independent copies of a subgaussian random variable with mean zero and unit variance. It holds for every $\varepsilon > 0$ that*

$$\mathbb{P}\left(\sigma_{\min}(\boldsymbol{A}) \geq \varepsilon(\sqrt{m} - \sqrt{n-1})\right) \geq 1 - (c_1\varepsilon)^{m-n+1} - \exp\left(-c_2 m\right),$$

where $c_1, c_2 > 0$ are constants depending polynomially only on the subgaussian moment.

Next, we present a probabilistic bound on the deviation of the norm of a Gaussian random vector from its mean. This is a direct extension of (Vershynin, 2018, Theorem 5.2.2).

Lemma 14 *Let $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_d)$ be a Gaussian random vector and $\lambda_1, \ldots, \lambda_d > 0$ be constants. It holds for any $t > 0$ that*

$$\mathbb{P}\left(\left|\sqrt{\sum_{i=1}^d \lambda_i^2 x_i^2} - \sqrt{\sum_{i=1}^d \lambda_i^2}\right| \geq t + 2\lambda_{\max}\right) \leq 2\exp\left(-\frac{t^2}{2\lambda_{\max}^2}\right), \tag{62}$$

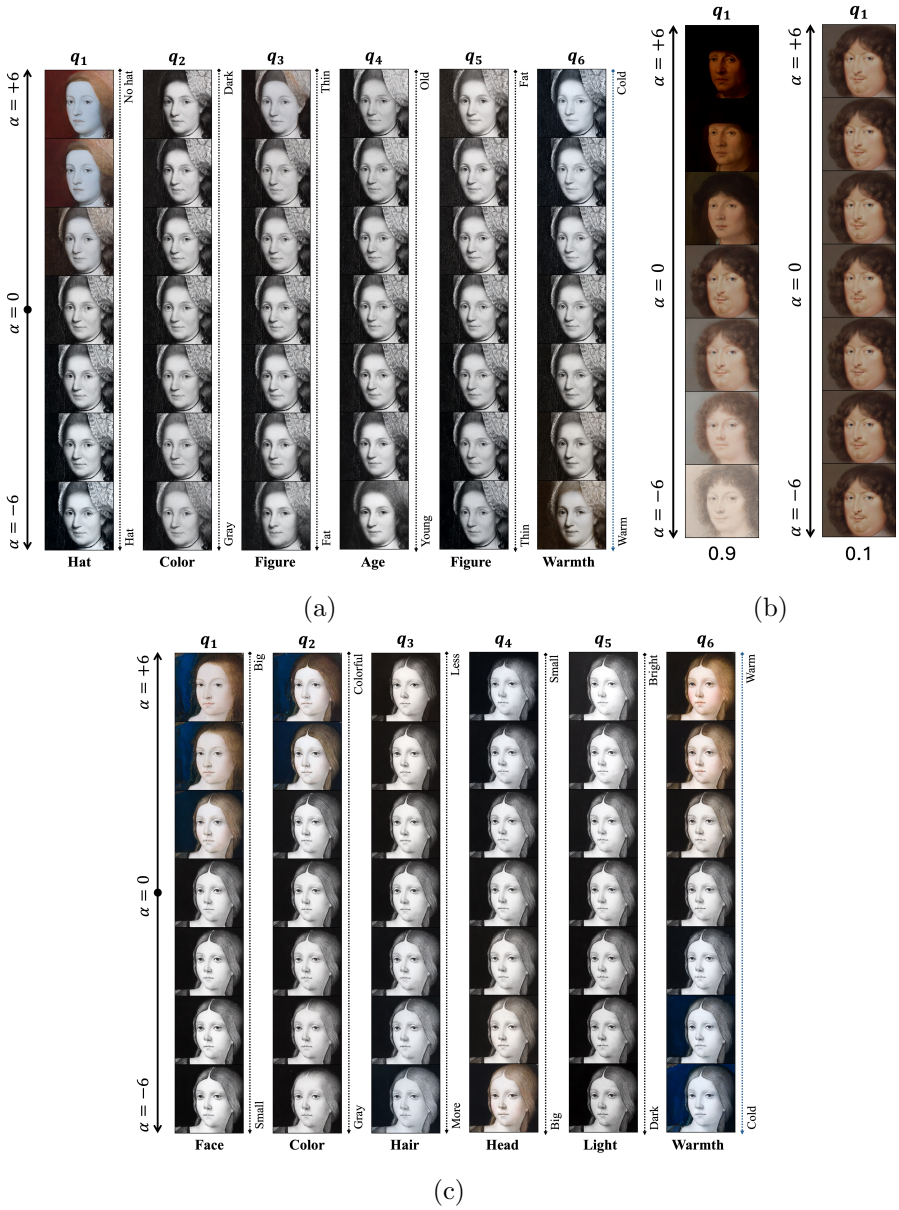


Figure 8: **Correspondence between the singular vectors of the Jacobian of the DAE and semantic image attributes.** (a,c) Additional examples when $t = 0.7$. (b) Ablation studies when $t = 0.1$ and 0.9.

where $\lambda_{\max} = \max\{\lambda_i : i \in [d]\}$.

Based on the above lemma, we can further show the following concentration inequalities to estimate the norm of the standard normal Gaussian random vector.

Lemma 15 *Suppose that* $\boldsymbol{a}_i \overset{i.i.d.}{\sim} \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_d)$ *is a Gaussian random vector for each* $i \in [N]$. *It holds for all* $i \in [N]$ *with probability at least* $1 - 2N^{-1}$ *that*

$$\left| \|\boldsymbol{a}_i\| - \sqrt{d} \right| \le 2\sqrt{\log N} + 2. \tag{63}$$

Proof Applying Lemma 14 to $\boldsymbol{a}_i \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_d)$, together with setting $t = 2\sqrt{\log N}$ and $\lambda_j = 1$ for all $j \in [d]$, yields

$$\mathbb{P}\left(\left| \|\boldsymbol{a}_i\| - \sqrt{d} \right| \ge 2\sqrt{\log N} + 2 \right) \le 2N^{-2}.$$

This, together with the union bound, yields that (63) holds with probability $1 - 2N^{-1}$. ■

Next, we present a spectral concentration bound for sample covariance matrices formed from arbitrary subsets of Gaussian random vectors; see Liu et al. (2026, Theorem 1).

Lemma 16 *Let* $\{\boldsymbol{a}_i\}_{i=1}^M \subseteq \mathbb{R}^d$ *be i.i.d. standard Gaussian random vectors. Fix* $\kappa \in (0, 1]$ *and* $\eta \in (0, 1/2)$, *and define*

$$\gamma := \Phi^{-1}\left(\frac{1 + (1-\eta)\kappa}{2} \right), \quad \mu := (1-\eta)\kappa - \sqrt{\frac{2\gamma^2}{\pi}} \exp\left(-\frac{\gamma^2}{2} \right).$$

Then there exists an absolute constant $c > 0$ *such that, with probability at least*

$$1 - 2\left(\frac{22}{\eta\sqrt{\mu}} + 1 \right)^d \left(\exp\left(-\frac{\eta^2 \kappa M}{2} \right) + \exp\left(-c\eta^2\mu^2 M \right) \right) - 2\exp(-2M),$$

it holds for every subset $\Omega \subseteq [M]$ *satisfying* $|\Omega| \ge \kappa M$ *that*

$$\lambda_{\min}\left(\sum_{i \in \Omega} \boldsymbol{a}_i \boldsymbol{a}_i^T \right) \ge (1-\eta)^2 \mu M.$$

Lemma 17 *Let* $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{n \times n}$ *be positive semi-definite matrices. Then, it holds that*

$$\langle \boldsymbol{A}, \boldsymbol{B} \rangle \ge \lambda_{\min}(\boldsymbol{A}) \mathrm{Tr}(\boldsymbol{B}). \tag{64}$$

Proof Let $\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T = \boldsymbol{A}$ be an eigenvalue decomposition of $\boldsymbol{A}$, where $\boldsymbol{U} \in \mathcal{O}^n$ and $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$ is a diagonal matrix with diagonal entries $\lambda_1 \ge \cdots \ge \lambda_n \ge 0$ being the eigenvalues. Then, we compute

$$\langle \boldsymbol{A}, \boldsymbol{B} \rangle = \langle \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T, \boldsymbol{B} \rangle = \langle \boldsymbol{\Lambda}, \boldsymbol{U}\boldsymbol{B}\boldsymbol{U}^T \rangle \ge \lambda_{\min}(\boldsymbol{A}) \mathrm{Tr}(\boldsymbol{U}\boldsymbol{B}\boldsymbol{U}^T) = \lambda_{\min}(\boldsymbol{A}) \mathrm{Tr}(\boldsymbol{B}),$$

where the inequality follows from $\lambda_i \ge 0$ for all $i \in [n]$ and $\boldsymbol{B}$ is a positive semidefinite matrix. ■

References

Pankaj K Agarwal and Nabil H Mustafa. K-means projective clustering. In *Proceedings of the 23rd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pages 155–165, 2004.

Sanjeev Arora, Rong Ge, Yingyu Liang, Tengyu Ma, and Yi Zhang. Generalization and equilibrium in generative adversarial nets (GANs). In *International Conference on Machine Learning*, pages 224–232. PMLR, 2017.

Iskander Azangulov, George Deligiannidis, and Judith Rousseau. Convergence of diffusion models under the manifold hypothesis in high-dimensions. *arXiv preprint arXiv:2409.18804*, 2024.

Omer Bar-Tal, Hila Chefer, Omer Tov, Charles Herrmann, Roni Paiss, Shiran Zada, Ariel Ephrat, Junhwa Hur, Guanghui Liu, Amit Raj, et al. Lumiere: A space-time diffusion model for video generation. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11, 2024.

Joe Benton, Valentin De Bortoli, Arnaud Doucet, and George Deligiannidis. Nearly d-linear convergence bounds for diffusion models via stochastic localization. In *International Conference on Learning Representations*, 2024.

Giulio Biroli, Tony Bonnaire, Valentin De Bortoli, and Marc Mézard. Dynamical regimes of diffusion models. *Nature Communications*, 15(1):9957, 2024.

Valentin De Bortoli. Convergence of denoising diffusion models under the manifold hypothesis. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856.

Bradley CA Brown, Anthony L Caterini, Brendan Leigh Ross, Jesse C Cresswell, and Gabriel Loaiza-Ganem. Verifying the union of manifolds hypothesis for image data. In *International Conference on Learning Representations*, 2023.

Hongrui Chen, Holden Lee, and Jianfeng Lu. Improved analysis of score-based generative modeling: User-friendly bounds under minimal smoothness assumptions. In *International Conference on Machine Learning*, pages 4735–4763. PMLR, 2023a.

Minshuo Chen, Kaixuan Huang, Tuo Zhao, and Mengdi Wang. Score approximation, estimation and distribution recovery of diffusion models on low-dimensional data. In *International Conference on Machine Learning*, pages 4672–4712. PMLR, 2023b.

Minshuo Chen, Renyuan Xu, Yumin Xu, and Ruixun Zhang. Diffusion factor models: Generating high-dimensional returns with factor structure. *arXiv preprint arXiv:2504.06566*, 2025a.

Sitan Chen, Sinho Chewi, Jerry Li, Yuanzhi Li, Adil Salim, and Anru R Zhang. Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions. In *International Conference on Learning Representations*, 2023c.

Sitan Chen, Vasilis Kontonis, and Kulin Shah. Learning general Gaussian mixtures with efficient score matching. In *The Thirty Eighth Annual Conference on Learning Theory*, pages 1029–1090. PMLR, 2025b.

Siyi Chen, Huijie Zhang, Minzhe Guo, Yifu Lu, Peng Wang, and Qing Qu. Exploring low-dimensional subspace in diffusion models for controllable image editing. In *Advances in Neural Information Processing Systems*, volume 37, pages 27340–27371, 2024.

Tianyu Chen, Yasi Zhang, Zhendong Wang, Ying Nian Wu, Oscar Leong, and Mingyuan Zhou. Denoising score distillation: From noisy diffusion pretraining to one-step high-quality generation. *arXiv preprint arXiv:2503.07578*, 2025c.

Xinlei Chen, Zhuang Liu, Saining Xie, and Kaiming He. Deconstructing denoising diffusion models for self-supervised learning. In *International Conference on Learning Representations*, volume 2025, pages 55458–55472, 2025d.

Zhengdao Chen. On the interpolation effect of score smoothing in diffusion models. In *The Fourteenth International Conference on Learning Representations*, 2026.

Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8188–8197, 2020.

Hyungjin Chung, Jeongsol Kim, Michael T McCann, Marc L Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations*, 2023.

Frank Cole and Yulong Lu. Score-based generative models break the curse of dimensionality in learning a family of sub-Gaussian distributions. In *International Conference on Learning Representations*, 2024.

Hugo Cui, Cengiz Pehlevan, and Yue M Lu. A precise asymptotic analysis of learning diffusion models: theory and insights. *arXiv preprint arXiv:2501.03937*, 2025.

Li Deng. The MNIST database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29:141–142, 2012.

Benjamin Dupuis, Dario Shariatian, Maxime Haddouche, Alain Oliviero Durmus, and Umut Simsekli. Algorithm- and data-dependent generalization bounds for diffusion models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011.

Ehsan Elhamifar and René Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11): 2765–2781, 2013.

Zalan Fabian, Berk Tinaz, and Mahdi Soltanolkotabi. Adapt and diffuse: Sample-adaptive reconstruction via latent diffusion models. *Proceedings of Machine Learning Research*, 235:12723, 2024.

Khashayar Gatmiry, Jonathan Kelner, and Holden Lee. Learning mixtures of Gaussians using diffusion models. In *Proceedings of Thirty Eighth Conference on Learning Theory*, volume 291, pages 2403–2456. PMLR, 2025.

Sixue Gong, Vishnu Naresh Boddeti, and Anil K Jain. On the intrinsic dimensionality of image representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3987–3996, 2019.

Georg A Gottwald, Shuigen Liu, Youssef Marzouk, Sebastian Reich, and Xin T Tong. Localized diffusion models for high-dimensional distributions generation. *arXiv preprint arXiv:2505.04417*, 2025.

Xiangming Gu, Chao Du, Tianyu Pang, Chongxuan Li, Min Lin, and Ye Wang. On memorization in diffusion models. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856.

Yinbin Han, Meisam Razaviyayn, and Renyuan Xu. Neural network-based score estimation in diffusion models: Optimization and generalization. In *International Conference on Learning Representations*, 2024.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851, 2020.

Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.

Zhihan Huang, Yuting Wei, and Yuxin Chen. Denoising diffusion probabilistic models are optimally adaptive to unknown low dimensionality. *Mathematics of Operations Research*, 2026.

Zahra Kadkhodaie, Florentin Guth, Eero P. Simoncelli, and Stephane Mallat. Generalization in diffusion models arises from geometry-adaptive harmonic representations. In *International Conference on Learning Representations*, 2024.

Hamid Kamkari, Brendan Ross, Rasa Hosseinzadeh, Jesse Cresswell, and Gabriel Loaiza-Ganem. A geometric view of data complexity: Efficient local intrinsic dimension estimation with diffusion models. In *Advances in Neural Information Processing Systems*, volume 37, pages 38307–38354, 2024.

Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In *Advances in Neural Information Processing Systems*, volume 33, pages 12104–12114, 2020.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Advances in Neural Information Processing Systems*, volume 35, pages 26565–26577, 2022.

Vahid Kazemi and Josephine Sullivan. One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1867–1874, 2014.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis. In *Advances in Neural Information Processing Systems*, volume 33, pages 17022–17033, 2020.

Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. DIFFWAVE: A versatile diffusion model for audio synthesis. In *International Conference on Learning Representations*, 2021.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *Technical Report*, 2009.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Holden Lee, Jianfeng Lu, and Yixin Tan. Convergence for score-based generative modeling with polynomial complexity. In *Advances in Neural Information Processing Systems*, volume 35, pages 22870–22882, 2022.

Holden Lee, Jianfeng Lu, and Yixin Tan. Convergence of score-based generative modeling for general data distributions. In *International Conference on Algorithmic Learning Theory*, pages 946–985. PMLR, 2023.

Gilad Lerman and Tyler Maunu. An overview of robust subspace recovery. *Proceedings of the IEEE*, 106(8):1380–1410, 2018.

Gilad Lerman, Kang Li, Tyler Maunu, and Teng Zhang. Global convergence of iteratively reweighted least squares for robust subspace recovery. *arXiv preprint arXiv:2506.20533*, 2025.

Gen Li and Changxiao Cai. Breaking AR's sampling bottleneck: Provable acceleration via diffusion language models. In *Advances in Neural Information Processing Systems*, volume 38, pages 11700–11725, 2026.

Gen Li and Yuantao Gu. Theory of spectral method for union of subspaces-based random geometry graph. In *International Conference on Machine Learning*, volume 139, pages 6337–6345. PMLR, 2021.

Gen Li and Yuling Yan. Adapting to unknown low-dimensional structures in score-based diffusion models. In *Advances in Neural Information Processing Systems*, 2024.

Gen Li, Yuting Wei, Yuejie Chi, and Yuxin Chen. A sharp convergence theory for the probability flow ODEs of diffusion models. *arXiv preprint arXiv:2408.02320*, 2024a.

Gen Li, Changxiao Cai, and Yuting Wei. Dimension-free convergence of diffusion models for approximate Gaussian mixtures. *arXiv preprint arXiv:2504.05300*, 2025a.

Puheng Li, Zhong Li, Huishuai Zhang, and Jiang Bian. On the generalization properties of diffusion models. *Advances in Neural Information Processing Systems*, 36:2097–2127, 2024b.

Runjia Li, Qiwei Di, and Quanquan Gu. Unified convergence analysis for score-based diffusion models with deterministic samplers. In *International Conference on Learning Representations*, 2025b.

Xiang Li, Yixiang Dai, and Qing Qu. Understanding generalizability of diffusion models requires rethinking the hidden Gaussian structure. In *Advances in Neural Information Processing Systems*, volume 37, pages 57499–57538, 2024c.

Jiadong Liang, Zhihan Huang, and Yuxin Chen. Low-dimensional adaptation of diffusion models: Convergence in total variation. In *Proceedings of Thirty Eighth Conference on Learning Theory*, volume 291, pages 3723–3729. PMLR, 2025.

Haixia Liu, Jian-Feng Cai, and Yang Wang. Subspace clustering by (k, k)-sparse matrix factorization. *Inverse Problems & Imaging*, 11(3), 2017.

Huikang Liu, Peng Wang, and Laura Balzano. A concentration inequality for the covariance matrix of an arbitrary subset of random vectors. *arXiv preprint arXiv:2606.24766*, 2026.

Yuanyuan Liu, LC Jiao, and Fanhua Shang. An efficient matrix factorization based low-rank representation for subspace clustering. *Pattern Recognition*, 46(1):284–292, 2013.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015.

Gabriel Loaiza-Ganem, Brendan Leigh Ross, Rasa Hosseinzadeh, Anthony L. Caterini, and Jesse C. Cresswell. Deep generative models through the lens of the manifold hypothesis: A survey and new connections. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

Yang Lyu, Yuchun Qian, Tan Minh Nguyen, and Xin T Tong. Resolving memorization in empirical diffusion model for manifold data in high-dimensional spaces. *arXiv preprint arXiv:2505.02508*, 2025.

Hila Manor and Tomer Michaeli. On the posterior distribution in denoising: Application to uncertainty quantification. In *International Conference on Learning Representations*, 2024a.

Hila Manor and Tomer Michaeli. Zero-shot unsupervised and text-based audio editing using DDPM inversion. In *International Conference on Machine Learning*, volume 235, pages 34603–34629. PMLR, 2024b.

Tyler Maunu, Teng Zhang, and Gilad Lerman. A well-tempered landscape for non-convex robust subspace recovery. *Journal of Machine Learning Research*, 20(37):1–59, 2019.

Claudia Merger and Sebastian Goldt. Generalization dynamics of linear diffusion models. *arXiv preprint arXiv:2505.24769*, 2025.

Kazusato Oko, Shunta Akiyama, and Taiji Suzuki. Diffusion models are minimax optimal distribution estimators. In *International Conference on Machine Learning*, pages 26517–26582. PMLR, 2023.

Ed Pizzi, Sreya Dutta Roy, Sugosh Nagavara Ravindra, Priya Goyal, and Matthijs Douze. A self-supervised descriptor for image copy detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14532–14542, 2022.

Phil Pope, Chen Zhu, Ahmed Abdelkader, Micah Goldblum, and Tom Goldstein. The intrinsic dimension of images and its impact on learning. In *International Conference on Learning Representations*, 2020.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, part III 18*, pages 234–241. Springer, 2015.

Mark Rudelson and Roman Vershynin. Smallest singular value of a random rectangular matrix. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 62(12):1707–1739, 2009.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115:211–252, 2015.

Kulin Shah, Sitan Chen, and Adam Klivans. Learning mixtures of Gaussians using the DDPM objective. In *Advances in Neural Information Processing Systems*, volume 36, pages 19636–19649, 2023.

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

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

Mahdi Soltanolkotabi and Emmanuel J Candes. A geometric analysis of subspace clustering with outliers. *The Annals of Statistics*, 40(4):2195–2238, 2012.

Mahdi Soltanolkotabi, Ehsan Elhamifar, and Emmanuel J Candes. Robust subspace clustering. *Annals of Statistics*, 42(2):669–699, 2014.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion art or digital forgery? investigating data replication in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6048–6058, 2023a.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Understanding and mitigating copying in diffusion models. In *Advances in Neural Information Processing Systems*, volume 36, pages 47783–47803, 2023b.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2020.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021a.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021b.

Jan Pawel Stanczuk, Georgios Batzolis, Teo Deveney, and Carola-Bibiane Schönlieb. Diffusion models encode the intrinsic dimension of data manifolds. In *International Conference on Machine Learning*, 2024.

Rong Tang and Yun Yang. Adaptivity of diffusion models to manifold structures. In *International Conference on Artificial Intelligence and Statistics*, pages 1648–1656. PMLR, 2024.

Berk Tinaz, Zalan Fabian, and Mahdi Soltanolkotabi. Emergence and evolution of interpretable concepts in diffusion models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.

Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

René Vidal. Subspace clustering. *IEEE Signal Processing Magazine*, 28(2):52–68, 2011.

René Vidal, Yi Ma, and Shankar Sastry. Generalized principal component analysis (GPCA). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1945–1959, 2005.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011.

Binxu Wang and John Vastola. The unreasonable effectiveness of Gaussian score approximation for diffusion models and its applications. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

Binxu Wang and John J Vastola. The hidden linear structure in score-based models and its application. *arXiv preprint arXiv:2311.10892*, 2023.

Peng Wang, Huikang Liu, Anthony Man-Cho So, and Laura Balzano. Convergence and recovery guarantees of the K-subspaces method for subspace clustering. In *International Conference on Machine Learning*, pages 22884–22918. PMLR, 2022.

Yu-Xiang Wang and Huan Xu. Noisy sparse subspace clustering. In *International Conference on Machine Learning*, pages 89–97. PMLR, 2013.

Zhendong Wang, Yifan Jiang, Huangjie Zheng, Peihao Wang, Pengcheng He, Zhangyang Wang, Weizhu Chen, Mingyuan Zhou, et al. Patch diffusion: Faster and more data-efficient training of diffusion models. *Advances in Neural Information Processing Systems*, 36:72137–72154, 2023a.

Zhixin Wang, Ziying Zhang, Xiaoyun Zhang, Huangjie Zheng, Mingyuan Zhou, Ya Zhang, and Yanfeng Wang. Dr2: Diffusion-based robust degradation remover for blind face restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1704–1713, 2023b.

Per-Åke Wedin. Perturbation bounds in connection with singular value decomposition. *BIT Numerical Mathematics*, 12:99–111, 1972.

Yuxin Wen, Yuchen Liu, Chen Chen, and Lingjuan Lyu. Detecting, explaining, and mitigating memorization in diffusion models. In *International Conference on Learning Representations*, 2023.

Andre Wibisono, Yihong Wu, and Kaylee Yingxi Yang. Optimal score estimation via empirical bayes smoothing. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 4958–4991. PMLR, 2024.

Yuchen Wu, Minshuo Chen, Zihao Li, Mengdi Wang, and Yuting Wei. Theoretical insights for diffusion guidance: A case study for Gaussian mixture models. In *International Conference on Machine Learning*, pages 53291–53327. PMLR, 2024.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms, 2017.

Zhen Xing, Qijun Feng, Haoran Chen, Qi Dai, Han Hu, Hang Xu, Zuxuan Wu, and Yu-Gang Jiang. A survey on video diffusion models. *ACM Computing Surveys*, 57(2):1–42, 2024.

Konstantin Yakovlev and Nikita Puchkin. Generalization error bound for denoising score matching under relaxed manifold assumption. In *Proceedings of Thirty Eighth Conference on Learning Theory*, volume 291, pages 5824–5891. PMLR, 2025.

TaeHo Yoon, Joo Young Choi, Sehyun Kwon, and Ernest K Ryu. Diffusion probabilistic models generalize when they fail to memorize. In *ICML 2023 Workshop on Structured Probabilistic Inference & Generative Modeling*, 2023.

Huijie Zhang, Jinfan Zhou, Yifu Lu, Minzhe Guo, Peng Wang, Liyue Shen, and Qing Qu. The emergence of reproducibility and consistency in diffusion models. In *International Conference on Machine Learning*, volume 235, pages 60558–60590. PMLR, 2024.