Diffusion Model's Generalization Can Be Characterized by Inductive Biases toward a Data-Dependent Ridge Manifold

Ye He [1] **Yitong Qiu** [2] **Molei Tao** [1]

Abstract

We study a data-dependent notion of diffusion-model generalization: when a model does not memorize the training set, where do its generated samples go relative to the geometry induced by the data? To answer this, we introduce a time-dependent family of log-density ridge manifolds constructed from the smoothed empirical distribution, and use it to characterize reverse-time inference. Our main result shows that generated samples evolve by a **reach–align–slide** mechanism: they first enter a neighborhood of the ridge, then their distance to the ridge is controlled by the normal component of training error, and finally their motion along the ridge is controlled by the tangential component. We further connect this geometric picture to training dynamics through directional decompositions of the learned error, and make this link explicit for random feature models, where architectural bias and optimization error can be separated quantitatively. Experiments on synthetic multimodal data and MNIST latent diffusion support the predicted geometric behavior in both low and high dimensions.

1. Introduction

Diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021) now achieve state-of-the-art sample quality across a wide range of generative tasks, making them a central tool for image, audio, and video generation (Dhariwal & Nichol, 2021; Kong et al., 2021; Brooks et al., 2024). At the same time, it is increasingly important to understand how innovative these models actually are, that is, what they generate beyond the training data. A central concern is memorization: in some regimes, diffusion models can behave like stochastic parrots that simply reproduce training data, raising both scientific and practical concerns, including privacy and safety risks (Carlini et al., 2023; Somepalli et al., 2023; Duan et al., 2023; Liu et al., 2024). Recent theory and empirical evidence suggest that non-memorizing behavior arises from various sources of error inside the learned diffusion model (e.g., Ye et al., 2025), and in literature the term *generalization* is often used in precisely this sense of non-memorization (e.g., Kadkhodaie et al., 2023; Zhang et al., 2023).

This leads to the central question of the paper:

When a diffusion model does not memorize the training set, where do its generated samples go?

Equivalently, once non-memorization has occurred, what geometric structure organizes the new samples produced by reverse-time inference? Our goal is to answer this question explicitly and quantitatively. In particular, we seek not merely to say that generated samples differ from the training data, but to describe where they are located relative to the geometry induced by the data.

This question should be understood in a fully data-dependent sense. Rather than taking an unknown population distribution as the primary reference, we take the finite training dataset itself as the object that defines the relevant geometry, and ask how generation departs from that geometry. In this sense, our focus is different from classical population-level generalization: we are not primarily asking how close the generated distribution is to an unknown population law, but how the model generates new samples relative to the observed data. This viewpoint is especially natural when one wants to understand structured intermediate generations between training samples, since the key issue is not only distributional discrepancy, but also the spatial organization of generated samples. We defer a more detailed comparison with related notions of generalization to Appendix A.

Our answer is geometric. We construct a time-indexed family of log-density ridge manifolds from the smoothed empirical distribution, and use this family as the reference geometry for reverse-time inference. Relative to these ridges, generated samples follow a *reach–align–slide* mechanism: after an initial transient they enter a neighborhood of the ridge, then move toward it in normal directions, and finally

[1]School of Mathematics, Georgia Institute of Technology, GA, USA [2]School of the Gifted Young, University of Science and Technology of China, Anhui, China. Correspondence to: Ye He <yhe367@gatech.edu>.

Published as a paper at the 1st FoGen workshop, ICML 2026, Seoul, South Korea, 2026.

evolve along it in tangent directions. This perspective goes beyond only locating generation near a low-dimensional geometric object: while the reach and align stages explain where generation concentrates, the slide stage provides additional information about how generation is organized along that geometry through tangential motion relative to nearby data-induced centers. At the same time, this description remains appropriately partial: our analysis predicts tangential components toward nearby data, rather than fully characterizing the generated configuration inside the tangent space. Moreover, this geometric picture is directly tied to training: the normal component of training error governs how closely samples align with the ridge, while the tangential component governs how far they slide along it.

Figure 1 illustrates this picture on a simple semi-circular dataset. After an initial phase, the generated samples *reach* a neighborhood of the log-density ridge, which then becomes the natural reference set for the remainder of inference. They subsequently *align* toward the ridge in normal directions, although the residual distance need not vanish and is controlled by the normal component of training error. At the same time, they *slide* along the ridge in tangent directions toward the training data, and the extent of this sliding is controlled by the tangential component of training error. In this way, the reach–align–slide decomposition gives a geometric description not only of where non-memorizing generation occurs, but also of part of its internal organization along the data-induced geometry.

A particularly tractable setting in which this training-to-geometry link becomes explicit is random feature neural networks (RFNNs) trained by gradient descent. In that setting, we derive directional decompositions of training error and show how approximation and optimization errors translate into quantitatively different alignment and sliding behaviors during inference. This RFNN analysis is not meant to model all practical architectures faithfully; rather, it serves as an explicit nonasymptotic example showing how architectural bias and training accuracy can jointly determine the geometry of diffusion generation. In this way, the RFNN case makes concrete the broader message of the paper: training affects generation through direction-dependent geometric effects relative to the ridge family.

Our perspective is most closely related to recent theoretical work that seeks to explain why diffusion models generate non-memorizing samples. One line of work studies whether generalization can already arise from the stochasticity or structure of the finite training target itself (Vastola, 2025; Bertrand et al., 2025); in contrast, we take the empirical training set as given and ask how the learned model generates relative to the geometry induced by that set. Another closely related direction analyzes training-induced bias, focusing on how model class, feature learning, or optimization shape the learned scores (Kamb & Ganguli, 2024; Shah et al., 2025; Wu et al., 2025; Bonnaire et al., 2025). Our contribution is complementary: rather than only asking what bias training creates, we quantify how that bias appears during inference through distinct normal and tangential effects relative to a data-dependent ridge family. Finally, several recent works study inference-time behavior under structured settings or geometry-adaptive smoothing (Baptista et al., 2025; Farghly et al., 2025; Li et al., 2025). Our work is closest in spirit to this direction, but differs in three ways. First, we make the relevant geometric object explicit from the empirical data through a time-dependent ridge family. Second, we characterize reverse-time inference relative to this geometry in a way that goes beyond concentration near a low-dimensional set: the reach–align–slide analysis also captures part of the tangential organization of generation relative to nearby data. Third, we connect this geometry back to directional components of training error.

Taken together, this paper connects training architecture and optimization choices to the data-dependent generation geometry of diffusion models in three steps: we

(1) introduce a time-dependent log-density ridge geometry induced by the smoothed empirical distribution, which provides the reference object for describing where generation occurs (Section 3.1).

(2) show that reverse-time inference evolves relative to this geometry by a reach–align–slide mechanism: the reach and align stages explain concentration toward the ridge, while the slide stage captures tangential organization relative to nearby data, yielding a geometric description of non-memorizing generation beyond mere concentration near a low-dimensional set (Sections 3.2–3.4).

(3) connect these geometric behaviors back to training by identifying directional error components that control alignment and sliding, and make this link explicit in a nonasymptotic RFNN+GD setting through architecture- and optimization-driven decompositions (Sections 4.1–4.2).

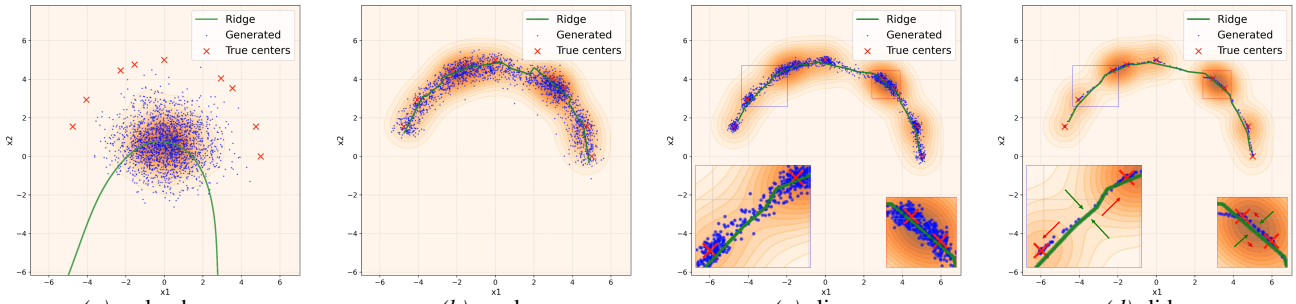
Empirical support for this training-to-geometry picture is provided in Section 5.

2. Preliminaries

SDE-based Diffusion Models. To generate samples in $\mathbb{R}^d$ from data samples $x_0^{(1)}, \cdots, x_0^{(n)} \in \mathbb{R}^d$, we consider the variance-preserving (VP) forward process $\mathrm{d}X_t = -X_t \mathrm{d}t + \sqrt{2}\mathrm{d}B_t$ for all $t \in [0, T]$, with marginals $p_t = \mathrm{Law}(X_t)$ and $p_0 = p$. The corresponding reverse process is

$$\mathrm{d}Y_t = \big(Y_t + 2\nabla \log p_{T-t}(Y_t)\big)\mathrm{d}t + \sqrt{2}\mathrm{d}\bar{B}_t. \qquad (1)$$

Write $a_t := e^{-t}$, $h_t = 1 - e^{-2t}$ so that $X_t = a_t X_0 + \sqrt{h_t} z$ in distribution with $z \sim \mathcal{N}(o, I_d)$ independent to X_0. We



(a) early phase *(b)* reach *(c)* align *(d)* slide

Figure 1. **Reach-align-slide on semi-circular dataset:** 9 unevenly spaced data points (red crosses) lie on a semi-circle of radius 4 centered at the origin. Generated samples (blue dots) evolve relative to the log-density ridge (green curve), exhibiting the reach–align–slide pattern. The zoom-in boxes in (c),(d) show region-dependent sliding: red arrows denote sliding directions and green arrows denote directions of continuation of alignment phase, with arrow lengths indicating intensity.

introduce the early stopping time $0 < \delta \ll 1$ so that the score $\nabla \log p_t$ is used only on $[\delta, T]$.

Denoising Mean Matching Loss. We consider learning the posterior mean instead of the score. By Tweedie's formula,

$$\nabla \log p_t(x) = -\frac{x}{h_t} + \frac{1}{h_t}\mathbb{E}_{x_0 \sim p_{0|t}(\cdot|x)}[a_t x_0], \quad (2)$$

where $p_{0|t}(\cdot|x)$ denotes the law of X_0 given $X_t = x$. Define

$$m(t,x) := \mathbb{E}_{x_0 \sim p_{0|t}(\cdot|x)}[a_t x_0], \quad (3)$$

Thus learning the score is equivalent to learning the posterior mean, which we use throughout the paper. We measure approximation error through the mean matching loss

$$\mathcal{L}_{\mathrm{MM}} = \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}\big[\|m_A(t,X_t) - m(t,X_t)\|^2\big]\mathrm{d}t, \quad (4)$$

where $m_A(t,x)$ is the learned posterior mean. In practice we train using its denoising version

$$\mathcal{L}_{\mathrm{DMM}} = \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}\big[\|-a_t X_0 + m_A(t,X_t)\|^2\big]\mathrm{d}t. \quad (5)$$

More details on $\mathcal{L}_{\mathrm{DMM}}$ are discussed in Appendix B. The simulated reverse process, initialized at Gaussian, is

$$\mathrm{d}\tilde{Y}_t = \big(\tilde{Y}_t + \frac{2(m_A(T-t,\tilde{Y}_t) - \tilde{Y}_t)}{h_{T-t}}\big)\mathrm{d}t + \sqrt{2}\mathrm{d}\tilde{B}_t. \quad (6)$$

Random Feature Neural Network and Gradient Descent. We parametrize the posterior mean by a RFNN:

$$m_A(t,x) := \frac{A}{\sqrt{p}}\sigma\Big(\frac{W_{\mathrm{x}}x}{\sqrt{d}} + \frac{W_{\mathrm{t}}\varphi_t}{\sqrt{2K_t+1}} + b\Big) := \frac{A}{\sqrt{p}}\sigma_t(x)$$

where $W_{\mathrm{x}} \in \mathbb{R}^{p\times d}, W_{\mathrm{t}} \in \mathbb{R}^{p\times(2K_{\mathrm{t}}+1)}$ are Gaussian random matrices. $\varphi_t \in \mathbb{R}^{2K_{\mathrm{t}}+1}$ consists of Fourier basis on $[0,T]$ and $b \sim \mathcal{N}(0, I_p)$ is the bias feature. σ is the activation function and $A \in \mathbb{R}^{d\times p}$ is the trainable parameter matrix. The corresponding denoising mean matching loss is

$$\mathcal{L}_{\mathrm{DMM}}(A) = \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}[\|-a_t X_0 + \frac{A}{\sqrt{p}}\sigma_t(X_t)\|^2]\mathrm{d}t.$$

We optimize the loss $\mathcal{L}_{\mathrm{DMM}}(A)$ by gradient descent with constant learning rate η, which yields

$$A_{k+1} - A_k = -2\eta A_k \tilde{U} + 2\eta\tilde{V}, \quad (7)$$

with
$$\tilde{U} = \int_\delta^T \frac{w(t)}{h_t^2}\frac{\mathbb{E}[\sigma_t(X_t)\sigma_t(X_t)^\intercal]}{p}\mathrm{d}t \in \mathbb{R}^{p\times p},$$
$$\tilde{V} = \int_\delta^T \frac{w(t)}{h_t^2}\frac{\mathbb{E}_z[a_t X_0 \sigma_t(X_t)^\intercal]}{\sqrt{p}}\mathrm{d}t \in \mathbb{R}^{d\times p}.$$

3. Geometric Properties of the Inference Process

In this section, we introduce the data-dependent ridge geometry that characterizes non-memorizing generation and study reverse-time inference relative to it. This yields the three-stage picture from the introduction: generated samples first reach a neighborhood of the ridge, then align toward it in normal directions and slide along it in tangent directions. Throughout the paper, we work under the following empirical-data setting.

Assumption 3.1. Data points $\{x_0^{(i)}\}_{i=1}^n$ are well-separated and bounded, i.e., $\Delta := \min_{i\neq j}\|x_0^{(i)} - x_0^{(j)}\| > 0$ and $R := \max_i \|x_0^{(i)}\| < \infty$. The data distribution p is the *empirical* distribution of the data, i.e., $p = \frac{1}{n}\sum_{i=1}^n \delta_{x_0^{(i)}}$.

3.1. Data-dependent Manifolds - Log-density Ridge Sets

For the smoothed empirical distribution p_t, log-density ridge is a geometric object that reflects the structure induced by the training data and serves as the reference object for describing generation; see Figure 1 for a visual example. Intuitively, it is a low-dimensional set along which the log-density is locally flat in normal directions and curved downward away from the set. In this sense, ridges generalize local modes: 0-dimensional ridges are isolated modes, 1-dimensional ridges trace connecting curves, and higher-dimensional ridges capture broader structures in the data.

Definition 3.2 (Log-density Ridge Sets)**.** For any smooth probability density $p \in \mathcal{P}(\mathbb{R}^d)$ and any positive integer $d^* < d$, the d^*-dimensional log-density ridge set of p with threshold $\beta > 0$, denoted as $\mathcal{R}_{d^*}(p;\beta)$, is defined by

$$\left\{x \in \mathbb{R}^d | E(x)E(x)^\intercal \nabla \log p(x) = 0, \lambda_{d^*+1}(x) \le -\beta\right\}$$

where $E(x) = (v_{d^*+1}(x), \cdots, v_d(x)) \in \mathbb{R}^{d\times(d-d^*)}$ with $\{(\lambda_i(x), v_i(x))\}_{i=1}^d$ being the eigenvalues/eigenvectors of $\nabla^2 \log p(x)$ in descending order, i.e., $\lambda_1(x) \ge \lambda_2(x) \cdots \ge \lambda_{d^*}(x) > \lambda_{d^*+1}(x) \ge \cdots \ge \lambda_d(x)$, for all $x \in \mathbb{R}^d$.

The definition identifies a d^*-dimensional set on which the log-density is locally maximal in the normal directions given by the bottom $(d-d^*)$-eigenspace of $\nabla^2 \log p(x)$: the condition $E(x)E(x)^\intercal \nabla \log p(x) = 0$ requires stationarity along them; the threshold $\lambda_{d^*+1}(x) \le -\beta$ enforces sufficient concavity in those directions, making the ridge geometrically identifiable at scale β.

Classical density ridges have been studied in statistics and geometry (Genovese et al., 2014; Chen et al., 2015). Here we instead introduce the ridge in *log-density space*, which is more natural for diffusion models: the reverse-time dynamics is governed by the score $\nabla \log p_t$, and the local geometry relevant for our analysis is determined by $\nabla^2 \log p_t$. For this reason, log-density ridges are better suited than classical density ridges for describing diffusion generation.

For each $t \in [\delta, T]$, we apply Definition 3.2 to forward marginal p_t and denote the ridge by $\mathcal{R}_t$. This yields a family of ridges varying with the noise level, which serves as the evolving reference geometry for reverse-time inference. The threshold is chosen at scale $\beta_t = \Theta(1/h_t)$, the natural curvature scale near the data, and also the scale needed for the later alignment estimates. See Remark D.6 for details.

Tube neighborhood and projection map. To analyze inference relative to $\mathcal{R}_t$, we need a notion of distance to the ridge (denoted by $\mathrm{dist}(\cdot, \mathcal{R}_t)$). For this purpose, we work in a tube neighborhood around $\mathcal{R}_t$ where the nearest-point projection onto the ridge is well-defined. This in turn requires geometric regularity of the ridge family, which we express through the following smoothness-and-reach assumption.

Assumption 3.3 (Smoothness and positive reach)**.** For any $t \in [\delta, T]$, there exists $r_t > 0$ such that $\mathcal{R}_t$ is a (piecewise) C^2-embedded submanifold in $\mathbb{R}^d$ with a reach no smaller than $r_t > 0$, i.e., for all $x \in \mathbb{R}^d$ with $\mathrm{dist}(x, \mathcal{R}_t) \le r_t$, there exists a unique nearest point on $\mathcal{R}_t$.

The following proposition provides the projection estimates needed for the later dynamical analysis.

Proposition 3.4. *Under Assumption 3.1, the log-density ridge family $\{\mathcal{R}_t\}_{\delta\le t\le T}$ satisfies Assumption 3.3. More precisely, as $t \to \delta^+ \ll 1$, the reach satisfies $r_t = \Omega(h_t^2\theta_t^{-1}R^{-3})$ for arbitrary $\theta_t = \exp(-o(h_t^{-1}))$. For any radius $\rho_t \in (0, r_t)$, define the tube neighborhood*

$$\mathcal{T}_t(\rho_t) := \{x \in \mathbb{R}^d | \mathrm{dist}(x, \mathcal{R}_t) \le \rho_t\}. \tag{8}$$

Then the nearest-point projection $\Pi_t : \mathcal{T}_t(\rho_t) \to \mathcal{R}_t$ is well-defined, and

(1) for all $x \in \mathcal{T}_t(\rho_t)$, the displacement $n_t(x) := x - \Pi_t(x)$ lies in the normal space of $\mathcal{R}_t$ at $\Pi_t(x)$;

(2) Π_t is C^1 on $\mathcal{T}_t(\rho_t)$ and $\sup_{x\in\mathcal{T}_t(\rho_t)} \|\nabla\Pi_t(x)\| \le \frac{1}{1-\rho_t/r_t}$;

(3) if $\rho_t = \Theta(r_t)$, the ridge motion is uniformly bounded: $\sup_{x\in\mathcal{T}_t(\rho_t)} \|\partial_t\Pi_t(x)\| = \mathcal{O}(R)$.

In particular, this proposition gives a well-defined projection onto the ridge inside a tube neighborhood, identifies projection residuals as normal directions, and controls both the spatial stability of the projection map and the time variation of the ridge family.

3.2. Stage 1 - Reaching the Tube Neighborhood

We first ask whether the inference trajectory enters the ridge tube, since the later normal/tangent analysis is meaningful only after projection becomes well-defined. We therefore introduce the first entrance time into the tube neighborhood and ask whether it occurs before inference ends.

Define $\tilde{t}_{\mathrm{in}} := \inf\{0 \le t \le T-\delta \mid \tilde{Y}_t \in \mathcal{T}_{T-t}(\rho_{T-t})\}$. The following theorem shows that, with high probability, the trajectory enters the tube before the end of inference.

Theorem 3.5 (Informal, formal one in Theorem E.1)**.** *Under Assumption 3.1, we have*

$$\mathbb{P}(\tilde{t}_{\mathrm{in}} \le T-\delta) \ge 1 - e_\delta - \varepsilon(T) - \sqrt{\varepsilon_A(T,\delta)/8},$$

where $\lim_{\delta\to 0^+} e_\delta = 0$, $\lim_{T\to\infty} \varepsilon(T) = 0$ and $\varepsilon_A(T,\delta) := \int_\delta^T h_t^{-2}\mathbb{E}[\|m(t,X_t) - m_A(t,X_t)\|^2]\mathrm{d}t$.

Thus, with high probability, the learned process reaches the ridge neighborhood before inference ends; the failure probability is controlled by early-stopping, large-time approximation, and global posterior-mean error.

3.3. Stage 2 - Aligning along Normal Directions

Once the inference trajectory enters the tube, we measure the off-ridge displacement through the squared normal distance to the ridge: $D_{T-t}(x) := \|x - \Pi_{T-t}(x)\|^2 = \|n_{T-t}(x)\|^2$. The key mechanism is contraction of this quantity along time after entry into the tube. In the main text, we state only the resulting bound at the final inference time $T-\delta$; the full time-resolved contraction estimate is given in Appendix F.

Theorem 3.6. *Under Assumption 3.1, let $e_A^\perp(t,x) := P^\perp(\Pi_t(x))e_A(t,x)$ with $e_A = m_A - m$. Choose $\beta_t = c/h_t$ for $c \in [\frac{1}{2}, 1)$[1]. Then for $\delta \ll 1$, $\mathbb{E}[D_\delta(\tilde{Y}_{T-\delta})]$ is of order*

[1] c can be chosen arbitrarily between $[\frac{1}{2}, 1)$ due to the property of $\nabla^2 \log p_t(x)$ as explained in Remark D.6.

$$\mathcal{O}\left(d\delta^c + \delta^c \int_{\tilde{t}_{\text{in}}}^{T-\delta} h_{T-u}^{-1-c}\, \mathbb{E}[\|e_A^{\perp}(T-u, \tilde{Y}_u)\|^2]\mathrm{d}u\right).$$

This theorem shows that the final squared distance to the ridge is controlled by a training-independent geometric term $d\delta^c$ and a cumulative contribution from the normal component of training residual. Thus, good normal alignment follows when the learned model has small error in directions transverse to the ridge.

3.4. Stage 3 - Sliding along Tangent Directions

The final stage concerns motion along the ridge rather than toward it. Near the end of inference, the smoothed empirical density p_{T-t} is a Gaussian mixture centered at the transported training points $\{m_{T-t}^{(i)} := a_{T-t}x_0^{(i)}\}_{i=1}^n$. In this regime, a trajectory typically enters a region where one mixture component is dominant. Inside such a region, the local tangent space of the ridge can be approximated using the top eigendirections of $\nabla^2 \log p_{T-t}$, which allows us to define tangent coordinates relative to the nearby center.

Define the i^{th} center-dominant region $\mathcal{B}_s^{(i)}(\theta_s) := \{x \in \mathbb{R}^d \mid \mathrm{Softmax}(-\frac{\|x-m_s\|^2}{2h_s})_i \geq 1-\theta_s\}$ where $\theta_s = \exp(-o(h_s^{-1}))$ as $s \to 0^+$. For $\tilde{Y}_t \in \mathcal{B}_{T-t}^{(i)}(\theta_{T-t})$, define the tangent coordinate

$$\tilde{u}_t^{(i)} := (U_{T-t}^{(i)})^{\intercal}(\tilde{Y}_t - m_{T-t}^{(i)}) \in \mathbb{R}^{d^*}, \tag{9}$$

where $U_{T-t}^{(i)} \in \mathbb{R}^{d\times d^*}$ consists of orthonormal columns as the top-d^* eigenvectors of $\nabla^2 \log p_{T-t}(m_{T-t}^{(i)})$. The vector $\tilde{u}_t^{(i)}$ measures the displacement of the sample along the local tangent directions of the ridge, relative to the nearby center $m_{T-t}^{(i)}$. As in the normal-direction analysis, the key mechanism is a time-evolution estimate along inference; in the main text we state only its consequence at the terminal time $T-\delta$. The full time-dependent estimate is deferred to Appendix G.

Theorem 3.7. *Under Assumption 3.1, let* $e_A^{\|,i}(t,x) = (U_t^{(i)})^{\intercal} e_A(t,x)$ *with* $e_A = m_A - m$. *If* $\tilde{Y}_t \in \mathcal{B}_{T-t}^{(i)}(\theta_{T-t})$, *then for* $\delta \ll 1$, $\mathbb{E}[\|\tilde{u}_{T-\delta}^{(i)}\|^2]$ *is of order*

$$\mathcal{O}\left(d\sqrt{\delta} + \sqrt{\delta} \int_{\tilde{t}_{\text{in}}}^{T-\delta} h_{T-u}^{-\frac{3}{2}} \mathbb{E}[\|e_A^{\|,i}(T-u, \tilde{Y}_u)\|^2]\mathrm{d}u\right).$$

This is the tangent analogue of Stage 2: the final amount of sliding along the ridge is controlled by a training-independent term $d\sqrt{\delta}$ and a cumulative contribution from the tangential training residual. Hence samples may align closely with the ridge without collapsing onto the training points, leaving structured intermediate generations.

Remark 3.8 (Effect of training weight on generation). Combining the normal and tangential bounds gives a simple interpretation of the role of training weight $w(t)$. Suppose the per-time contribution satisfies $w(t)h_t^{-2}\mathbb{E}[\|e_A(T-t, \tilde{Y}_t)\|^2] = \mathcal{O}(1)$, then the cumulative mean-error terms in the normal and tangential bounds scale as

$$\delta^c \int_{\delta}^{T-\tilde{t}_{\text{in}}} h_t^{1-c}/w(t)dt \quad \text{and} \quad \delta^{\frac{1}{2}} \int_{\delta}^{T-\tilde{t}_{\text{in}}} h_t^{\frac{1}{2}}/w(t)dt,$$

respectively. Hence placing larger weight on small t suppresses end-stage errors in both directions, which pushes the model toward memorization by reducing both off-ridge deviation and along-ridge spread. In particular, for $w(t) = h_t^2,\ h_t,\ 1$, the resulting mean-error scalings in both directions are $\mathcal{O}(1)$, $\mathcal{O}(\delta)$, and $\mathcal{O}(\delta^2)$, respectively, leading to different inductive biases in generation.

4. How Training Affects Generation: General Theory + an Explicit RF Example

Section 3 showed that non-memorizing generation is governed by directional errors: normal error controls alignment to the ridge, while tangential error controls spread along it. In this section, we connect these geometric quantities back to training. We first show that they can be controlled by corresponding directional components of the posterior mean matching loss, and then make this connection explicit in RFNN, where the same directional errors further split into architecture-driven and optimization-driven parts.

4.1. Directional Decomposition of Training Loss

Since the reverse-time dynamics depends linearly on the learned posterior mean, the posterior mean matching loss decomposes naturally into normal and tangential parts: $\mathcal{L}_{\text{MM}} = \mathcal{L}_{\text{MM}}^{\perp} + \mathcal{L}_{\text{MM}}^{\|}$ s.t. for $\dagger \in \{\perp, \|\}$,

$$\mathcal{L}_{\text{MM}}^{\dagger}(A) := \int_{\delta}^{T} \frac{w(t)\mathbb{E}[\|P_t^{\dagger}(X_t)e_A(t,X_t)\|^2]}{h_t^2}\mathrm{d}t$$

with $e_A := m_A - m$. $\mathcal{L}_{\text{MM}}^{\perp}$ measures training error in directions normal to the ridge and $\mathcal{L}_{\text{MM}}^{\|}$ measures training error in tangent directions.

Theorem 4.1. *Under mild assumptions, the normal and tangential errors in Theorems 3.6 and 3.7 can be estimated by projected posterior mean matching loss in corresponding directions:*

$$\text{normal-error bound} \lesssim C_\delta^{\perp}\mathcal{L}_{\text{MM}}^{\perp} + d\delta^c + C_\delta^{\perp}(\sqrt{d}+R)e^{-T},$$
$$\text{tangent-error bound} \lesssim C_\delta^{\|}\mathcal{L}_{\text{MM}}^{\|} + d\delta + C_\delta^{\|}(\sqrt{d}+R)e^{-T},$$

where $C_\delta^{\perp} := \delta^c\big(1 \vee \frac{\delta^{1-c}}{w(\delta)}\big)$, $C_\delta^{\|} := \delta^{\frac{1}{2}}\big(1 \vee \frac{\delta^{\frac{1}{2}}}{w(\delta)}\big)$ *and* $c = \lim_{t\to\delta} h_t\beta_t$ *is arbitrary in* $[\frac{1}{2}, 1)$.

Theorem 4.1, proved in Appendix H, shows that the geometric errors from Section 3 are controlled by projected training

losses, up to training-independent remainders that vanish as $\delta \to 0$ and $T \to \infty$. Thus normal training loss predicts alignment, while tangential training loss predicts sliding.

4.2. RFNN: Architecture and Optimization Effects on Generation

We now specialize this training-to-geometry connection to RFNN trained by gradient descent. In this setting, the directional training losses become explicit and can be further decomposed into two qualitatively different parts: an *architecture* term, reflecting the finite-width approximation floor, and an *optimization* term, reflecting incomplete training from initialization. This decomposition allows us to distinguish how model class and training procedure affect alignment and sliding.

Theorem 4.2. *Assume the conditions in Theorem 4.1 hold. Let $\{A_k\}_{k\geq 0}$ be the GD iterates in (7) with learning rate $\eta < \frac{2}{\lambda_1}$, then up to remainders controlled by δ, T,*

- *the normal error at training step k is bounded by $C_\delta^\perp(\mathrm{Err}_{arc}^\perp + \mathrm{Err}_{train}^\perp(k))$;*
- *the tangential error at training step k is bounded by $C_\delta^\parallel(\mathrm{Err}_{arc}^\parallel + \mathrm{Err}_{train}^\parallel(k))$.*

For each $\dagger \in \{\perp, \parallel\}$, $\mathrm{Err}_{arc}^\dagger$ is the architecture-driven term and $\mathrm{Err}_{train}^\dagger(k)$ is the optimization-driven term; their explicit formulas are given in Appendix I.

Theorem 4.2 makes the training-to-geometry connection explicit in RFNN by showing that both normal and tangential generation errors split into an architecture-driven part and an optimization-driven part. Thus, the ridge-based analysis does more than say that training matters: it separates how model class and training procedure contribute to generation, and does so differently in normal and tangential directions. The explicit form of this split, and its dependence on initialization and spectrum, becomes especially transparent in the two-point example below.

Fully Explicit Results of Two-point Data. WLOG assume that $x_0^{(1)} = (-\mu, 0)$ and $x_0^{(2)} = (\mu, 0)$. Then the ridge $\mathcal{R}_t \equiv \{x_2 = 0\}$ and the posterior mean $m(t, x) = (a_t\mu \tanh(\frac{a_t\mu}{h_t}x_1), 0)$. The initialization of GD writes as $A_0 = (A_{0,1}, A_{0,2})^\intercal$.

In this setting, the optimization-driven errors can be written explicitly, making the role of initialization and spectral bias fully visible: with $\{(\lambda_i, u_i)\}_{i=1}^r$ the spectrum of $\tilde{U}$ and $\tilde{v}$ the first row of $\tilde{V}$ for $\tilde{U}, \tilde{V}$ in (7),

$$\mathrm{Err}_{train}^\parallel(k) = \sum_{i=1}^{r} \lambda_i(1-2\eta\lambda_i)^{2k}(A_{0,1}^\intercal u_i - \tilde{v}^\intercal u_i/\lambda_i)^2$$
$$\mathrm{Err}_{train}^\perp(k) = \sum_{i=1}^{r} \lambda_i(1-2\eta\lambda_i)^{2k}(A_{0,2}^\intercal u_i)^2.$$

These expressions reveal a strong directional asymmetry. In particular, the normal optimization error depends only on the second row of the initialization. If that component vanishes, then $\mathrm{Err}_{train}^\perp(k) \equiv 0$, so the theory predicts essentially immediate alignment to the ridge. By contrast, if the initialization is aligned with the slowest spectral mode, then $\mathrm{Err}_{train}^\perp(k)$ decays only at the rate determined by the smallest positive eigenvalue, so normal alignment can remain poor for a long time.

In contrast, the architecture-driven error is purely tangential: $\mathrm{Err}_{arc}^\perp = 0$, while

$$\mathrm{Err}_{arc}^\parallel = \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}_{x\sim p_t}[a_t^2\mu^2 \tanh(\tfrac{a_t\mu}{h_t}x_1)^2]\mathrm{d}t - \tilde{v}^\intercal\tilde{U}^+\tilde{v}$$

is strictly positive at finite p. Hence samples can align strongly to the ridge while still spreading along it, producing edge-like interpolation between the two data points. As width increases, this tangential floor shrinks, and the behavior becomes increasingly memorization-like.

This example isolates the two core mechanisms of the paper: optimization error can delay normal alignment, especially through slow spectral modes, while finite-width architecture error can sustain tangential spreading along the ridge. Their combination yields non-memorizing generation with strong ridge alignment but persistent along-ridge spread. Numerical illustrations are given in Section 5.2.

5. Experiments

In this section, we evaluate the proposed geometric framework from three complementary perspectives. We begin with simple 2D examples that illustrate the role of the log-density ridge in characterizing non-memorizing generation. We then turn to the 2D two-point problem, where the ridge is explicit and the theoretical quantities can be checked quantitatively against the generated samples. Finally, we study MNIST latent diffusion and show that the same reach-align-slide picture remains informative in higher dimensions. Experimental details are deferred to Appendix J, and additional experiments are presented in Appendix K.

5.1. 2D Illustrations of the Role of Ridge

We first consider two simple 2D examples to illustrate that the ridge geometry explains generation even when the relevant low-dimensional structure is more complicated than a straight line. The goal here is not quantitative verification, but to show that the proposed log-density ridge captures nontrivial generation patterns from the data.

Our first example uses four training points at $(\pm 1, \pm 1)$. As shown in Figure 2, the generated samples concentrate along edge-like structures that are not part of the training set. The moving ridge tracks this behavior closely, indicating that it correctly predicts where non-memorizing generation occurs.

The second example shows that this phenomenon is not limited to straight-line interpolation. Here the training distribution is supported on the three points $(0, 0)$, $(3, 0)$, and $(0, 5)$. In Figure 3, the corresponding ridge is visibly bent, and the generated samples follow this curved geometry rather than concentrating on a straight segment between modes. This shows that the proposed ridge can capture genuinely curved low-dimensional structures induced by the data.

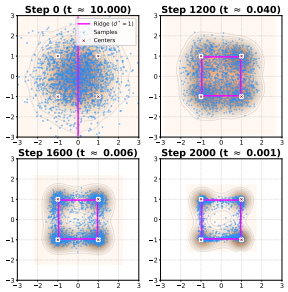


Figure 2. Generalization from 4 training points.

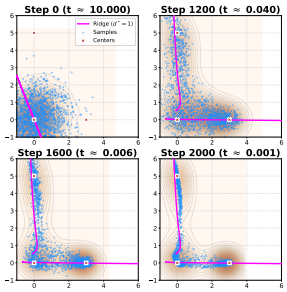
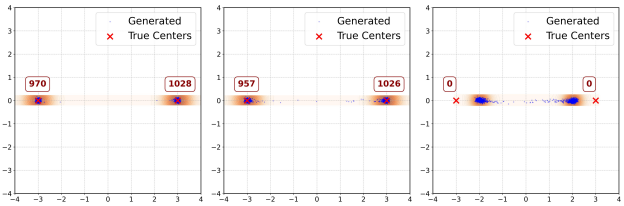


Figure 3. Generalization from 3 training points.

5.2. Synthetic Data - Two Points in 2D Plane

We next turn to the simplest setting in which the geometry is completely explicit:: a two-point dataset $\{(\pm 3, 0)\}$. In this case, the ridge is exactly the horizontal axis, so the tangent and normal directions are simply $e_1 = (1, 0)$ and $e_2 = (0, 1)$. This makes the example ideal for quantitatively testing both the geometric predictions of Section 3 and the training-to-geometry mechanism of Section 4. We use RFNN in this subsection; the MLP results show the same qualitative behavior and are deferred to Appendix K.

Directional geometry and its training origin. In this explicit two-point setting, we can directly compare the predicted directional quantities with the observed sample geometry. Figure 4 shows the generated samples under three weighting schedules $w(t) \in \{1, h_t, h_t^2\}$, while Figure 5(a) reports the corresponding normal and tangential geometric errors in Section 3 and Figure 5(b) reports the corresponding directional training losses in Section 4.

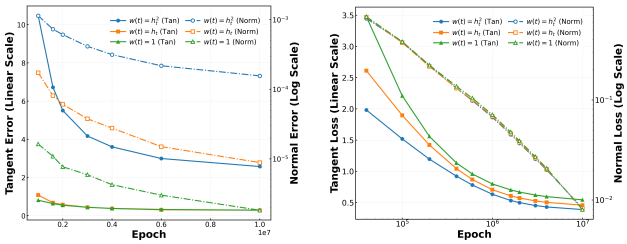


(a) $w(t) = 1$ (b) $w(t) = h_t$ (c) $w(t) = h_t^2$

Figure 4. **Generated samples with RFNN.** (a)–(c) Comparison of generated sample configurations under different weight schedules. Boxed numbers indicate sample counts around the target modes (radius $= 0.5$). The background color represents the KDE plot.

The geometric pattern is clear when Figure 4 is read together with Figure 5(a). Across all schedules, the normal error in Figure 5(a) remains extremely small, and correspondingly the generated samples in Figure 4 stay tightly concentrated near the ridge $y = 0$. The tangential configurations, however, differ substantially across schedules and are consistent with the tangential errors reported in Figure 5(a): for $w(t) = 1$, the tangential error is smallest and the samples are concentrated almost entirely around the two data points; for $w(t) = h_t^2$, the tangential error is largest and the samples exhibit a pronounced edge structure; the geometry for $w(t) = h_t$ lies between these two extremes. This is exactly the trend predicted by the theory: once normal alignment is achieved, the remaining tangential error determines how strongly the generated samples spread along the ridge.

Figure 5(b) explains where this behavior comes from at the training level. The normal loss remains small across all schedules, consistent with the uniformly strong normal alignment in Figure 5(a). The tangential behavior is more subtle and highlights why Theorem 4.1 is needed: $w(t) = h_t^2$ yields a relatively small tangential training loss while still producing a large tangential geometric effect, because the coefficient $C_\delta^{\parallel}$ strongly amplifies end-stage tangential error when $w(\delta)$ is small. Conversely, $w(t) = 1$ has a larger tangential training loss but still induce smaller tangential spread because the corresponding amplification factor is much weaker. Thus, the experiment validates Theorem 4.1: directional training losses control directional geometric errors, which in turn determine the observed sample geometry. This is also consistent with the effect of $w(t)$ predicted by Remark 3.8.

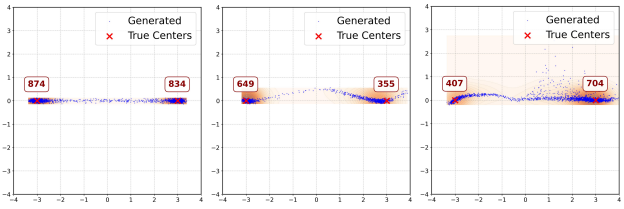


(a) Error Dynamics (b) Loss Decomposition

Figure 5. **Error dynamics and training loss decomposition of RFNN.** (a) Evolution of tangent- and normal- errors (b) Evolution of tangent- and normal- loss components. Both under weights $1, h_t, h_t^2$, with solid lines, linear scale on left axis for tangent direction; dash-dot lines, log scale on right axis for normal direction.

Biases of Different Initializations. We next test the initialization effects predicted by the RFNN analysis in Section 4.2. Figure 6 compares finite-training-time generation under three initialization schemes: zero, all-ones, and slow-spectrum. The zero initialization yields samples that remain essentially on the horizontal ridge, while the other two produce a visible arch before full alignment is reached.

This is consistent with the RFNN analysis in Section 4.2:



(a) Zero (b) Ones (c) Slow-Spec

Figure 6. **Initialization Effects (Epoch 40k).** Comparison of generated sample configurations under different initializations. The colored shading denotes the KDE of the distribution.

initialization mainly affects how quickly normal alignment is achieved. When the relevant slow modes are weak, alignment is nearly immediate; when they are emphasized, the approach to the ridge is much slower, producing the transient arch-shaped geometry seen in the figure.

5.3. Higher Dimension Data - MNIST

We next test whether the same geometric picture remains meaningful in higher dimensions. We study a binary MNIST problem in latent space and ask whether the generated samples exhibit the two behaviors predicted by the theory: normal alignment toward the ridge geometry and limited tangential sliding toward the training data.

Experimental setup. We consider the digits 4 and 8 from MNIST. To simplify the problem, we first train a VAE to embed the images into a 32-dimensional latent space, and then train a time-conditioned MLP score model in that latent space with weight $w(t) = 1$. Full architectural and training details are deferred to Appendix J.3.

Qualitative Visualization. Figure 7 provides a qualitative view of the generation dynamics through UMAP. At the beginning of inference, the generated samples are far from the ridge structure; as reverse-time inference proceeds, they move toward the global geometry captured by the ridge. This supports the ridge-based description of inference, although only at a qualitative level, since UMAP does not preserve the normal/tangent decomposition needed for a precise test of the theory.

Quantitative normal and tangential behavior. We next examine the two directional effects quantitatively. Since the true latent distribution is unknown, we estimate distance to the ridge by numerically solving the corresponding constrained optimization problem, with details in Appendix J.4. Figure 8(a) plots the mean distance to the ridge over 200 inference trajectories. The distance decreases steadily over most of inference and then stabilizes at a small floor near the end, matching the predicted normal-alignment stage.

Tangential motion behaves differently. Figure 8(b) shows the tangent error over the full inference horizon, and Figure 8(c) zooms in on the final stage. Compared to the normal distance, tangential motion decreases more slowly and

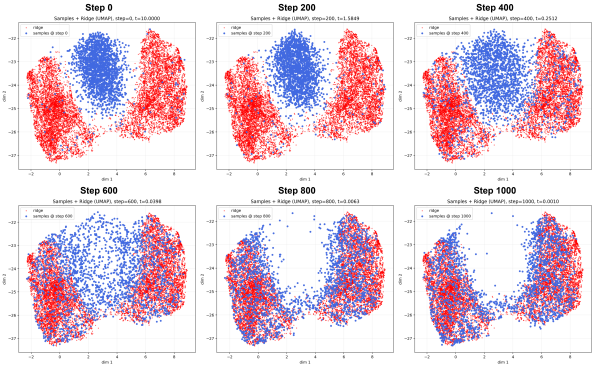
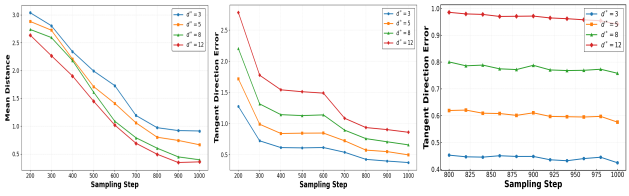


Figure 7. **UMAP visualization of generated samples and ridge.** Red points represent the underlying ridge structure $\mathcal{R}_t$ with $t = 0.001$, and blue points represent the generated samples at different time steps. The ridge accurately captures the sample distribution.

becomes negligible only near the end of inference. This temporal imbalance is the key observation: most of inference is spent aligning toward the ridge, leaving limited time for substantial sliding toward the exact training examples. As a result, the generated samples organize around the ridge without fully collapsing onto the training set, which is precisely the non-memorizing mechanism predicted by the theory.



(a) distance to ridge (b) tangent error(full) (c) tangent error(zoom)

Figure 8. **Evolution of Ridge Manifold metrics during inference.** (a) Mean distance versus sampling steps for d^* from step 200. (b) Tangential error trajectories from step 200. (c) Zoom-in of the tangential error in the final 200 steps.

6. Conclusions and Limitations

This work gives a data-dependent geometric perspective on diffusion models' generation. We show that when memorization does not occur, generated samples are organized by a time-dependent log-density ridge geometry induced by the training data, and that reverse-time inference follows a reach–align–slide mechanism relative to this geometry. The analysis also clarifies how training affects generation: normal training error controls alignment to the ridge, while tangential training error controls spread along it.

Our analysis has several limitations. We do not study errors induced by time discretization, which could alter both normal and tangential geometry, although prior discretization results suggest these effects should remain limited unless the step sizes are very large (Lee et al., 2022; De Bortoli, 2022; Chen et al., 2023b;a; Benton et al., 2024; Conforti et al., 2023; Wang et al., 2024). In addition, our explicit training-level decomposition is developed in the RFNN set-

ting, which serves as a tractable nonasymptotic example rather than a full model of modern diffusion architectures. Extending the present framework to discretized samplers and richer learning models would be natural next steps.

Impact Statement

This paper presents work whose goal is to advance the field of machine learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

Acknowledgment

MT thanks Mikhail Belkin and Peter L. Bartlett for insightful discussions.

References

Aamari, E., Kim, J., Chazal, F., Michel, B., Rinaldo, A., and Wasserman, L. Estimating the reach of a manifold. 2019.

Baptista, R., Dasgupta, A., Kovachki, N. B., Oberai, A., and Stuart, A. M. Memorization and regularization in generative diffusion models. *arXiv preprint arXiv:2501.15785*, 2025.

Benton, J., De Bortoli, V., Doucet, A., and Deligiannidis, G. Nearly d-linear convergence bounds for diffusion models via stochastic localization. In *The Twelfth International Conference on Learning Representations*, 2024.

Bertrand, Q., Gagneux, A., Massias, M., and Emonet, R. On the closed-form of flow matching: Generalization does not arise from target stochasticity. *arXiv preprint arXiv:2506.03719*, 2025.

Bonnaire, T., Urfin, R., Biroli, G., and Mézard, M. Why diffusion models don't memorize: The role of implicit dynamical regularization in training. *arXiv preprint arXiv:2505.17638*, 2025.

Brooks, T., Peebles, B., Holmes, C., DePue, W., Guo, Y., Jing, L., Schnurr, D., Taylor, J., Luhman, T., Luhman, E., et al. Video generation models as world simulators. *OpenAI Blog*, 1(8):1, 2024.

Carlini, N., Hayes, J., Nasr, M., Jagielski, M., Sehwag, V., Tramer, F., Balle, B., Ippolito, D., and Wallace, E. Extracting training data from diffusion models. In *32nd USENIX security symposium (USENIX Security 23)*, pp. 5253–5270, 2023.

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

Chen, S., Chewi, S., Li, J., Li, Y., Salim, A., and Zhang, A. R. Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions. *ICLR*, 2023b.

Chen, Y.-C., Genovese, C. R., and Wasserman, L. Asymptotic theory for density ridges. *The Annals of Statistics*, pp. 1896–1928, 2015.

Chen, Z. On the interpolation effect of score smoothing. *arXiv preprint arXiv:2502.19499*, 2025.

Conforti, G., Durmus, A., and Silveri, M. G. Score diffusion models without early stopping: finite fisher information is all you need. *arXiv preprint arXiv:2308.12240*, 2023.

De Bortoli, V. Convergence of denoising diffusion models under the manifold hypothesis. *TMLR*, 2022.

Dhariwal, P. and Nichol, A. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Duan, J., Kong, F., Wang, S., Shi, X., and Xu, K. Are diffusion models vulnerable to membership inference attacks? In *International Conference on Machine Learning*, pp. 8717–8730. PMLR, 2023.

Farghly, T., Potaptchik, P., Howard, S., Deligiannidis, G., and Pidstrigach, J. Diffusion models and the manifold hypothesis: Log-domain smoothing is geometry adaptive. *arXiv preprint arXiv:2510.02305*, 2025.

Genovese, C. R., Perone-Pacifico, M., Verdinelli, I., and Wasserman, L. Nonparametric ridge estimation. 2014.

George, A. J., Veiga, R., and Macris, N. Denoising score matching with random features: Insights on diffusion models from precise learning curves. *arXiv preprint arXiv:2502.00336*, 2025.

He, Y., Rojas, K., and Tao, M. Zeroth-order sampling methods for non-log-concave distributions: Alleviating metastability by denoising diffusion. *Advances in Neural Information Processing Systems*, 37:71122–71161, 2024.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Kadkhodaie, Z., Guth, F., Simoncelli, E. P., and Mallat, S. Generalization in diffusion models arises from geometry-adaptive harmonic representations. *arXiv preprint arXiv:2310.02557*, 2023.

Kamb, M. and Ganguli, S. An analytic theory of creativity in convolutional diffusion models. *arXiv preprint arXiv:2412.20292*, 2024.

Kong, Z., Ping, W., Huang, J., Zhao, K., and Catanzaro, B. Diffwave: A versatile diffusion model for audio synthesis. *ICLR*, 2021.

Lee, H., Lu, J., and Tan, Y. Convergence for score-based generative modeling with polynomial complexity. *Advances in Neural Information Processing Systems*, 35: 22870–22882, 2022.

Leobacher, G. and Steinicke, A. Existence, uniqueness and regularity of the projection onto differentiable manifolds. *Annals of global analysis and geometry*, 60(3):559–587, 2021.

Li, X., Shen, Z., Hsieh, Y.-P., and He, N. When scores learn geometry: Rate separations under the manifold hypothesis. *arXiv preprint arXiv:2509.24912*, 2025.

Liu, Y., Huang, J., Li, Y., Wang, D., and Xiao, B. Generative ai model privacy: a survey. *Artificial Intelligence Review*, 58(1):33, 2024.

Moshksar, K. Refining concentration for gaussian quadratic chaos. *arXiv preprint arXiv:2412.03774*, 2024.

Niyogi, P., Smale, S., and Weinberger, S. Finding the homology of submanifolds with high confidence from random samples. *Discrete & Computational Geometry*, 39(1): 419–441, 2008.

Shah, K., Kalavasis, A., Klivans, A. R., and Daras, G. Does generation require memorization? creative diffusion models using ambient diffusion. *arXiv preprint arXiv:2502.21278*, 2025.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pp. 2256–2265. pmlr, 2015.

Somepalli, G., Singla, V., Goldblum, M., Geiping, J., and Goldstein, T. Understanding and mitigating copying in diffusion models. *Advances in Neural Information Processing Systems*, 36:47783–47803, 2023.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. *ICLR*, 2021.

Vastola, J. J. Generalization through variance: how noise shapes inductive biases in diffusion models. *arXiv preprint arXiv:2504.12532*, 2025.

Wang, Y., He, Y., and Tao, M. Evaluating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems*, 37:19307–19352, 2024.

Wu, Y.-H., Marion, P., Biau, G., and Boyer, C. Taking a big step: Large learning rates in denoising score matching prevent memorization. *arXiv preprint arXiv:2502.03435*, 2025.

Ye, Z., Zhu, Q., Tao, M., and Chen, M. Provable separations between memorization and generalization in diffusion models. *arXiv preprint arXiv:2511.03202*, 2025.

Zhang, H., Zhou, J., Lu, Y., Guo, M., Wang, P., Shen, L., and Qu, Q. The emergence of reproducibility and consistency in diffusion models. *arXiv preprint arXiv:2310.05264*, 2023.

Contents

A. Related Work

This appendix expands the brief discussion in the introduction and clarifies how our notion of generalization and our ridge-based analysis relate to several nearby directions in the diffusion-model literature. Our goal is not to survey the entire area, but to explain more precisely which question our paper addresses and how it connects to existing viewpoints.

Population-level generalization versus our data-dependent question. A substantial line of work (e.g. Wang et al., 2024; Bertrand et al., 2025; Ye et al., 2025; Bonnaire et al., 2025) studies diffusion-model generalization by comparing the generated distribution to an unknown population distribution and deriving bounds in global discrepancy metrics. This perspective is natural when the goal is population-level recovery or distributional approximation. Our paper addresses a different question. We take the finite training dataset itself as the primary reference object and ask where generated samples go relative to the geometry induced by that dataset. In this sense, our focus is not primarily on global closeness to an unknown population law, but on the geometric organization of non-memorizing generations relative to the observed data. This viewpoint is especially useful when the phenomenon of interest is structured intermediate generation between training samples, since the relevant issue is not only how different two distributions are, but also how generated samples are spatially arranged.

Target-side stochasticity and finite-data target structure. One line of work (e.g., Vastola, 2025; Bertrand et al., 2025) asks whether generalization can already arise from the stochasticity or structure of the finite-data training target itself. From this viewpoint, the learned diffusion model may generate non-memorizing samples not only because of imperfections in training or inference, but also because the empirical target differs from a population-level object in a structured way. Our framework is related to this direction in that it also adopts a fully finite-data viewpoint. However, our emphasis is different: rather than analyzing the stochastic gap between empirical and population targets, we take the empirical dataset as given and study how reverse-time inference organizes samples relative to the geometry induced by that dataset.

Training-induced bias. A closely related direction studies the inductive bias created during training, for example through model class, feature learning, optimization dynamics, or finite training time (e.g., Kamb & Ganguli, 2024; Shah et al., 2025; Wu et al., 2025; Bonnaire et al., 2025). This literature explains how the learned score or posterior mean differs from the ideal one and how such differences depend on architecture and optimization. Our contribution is complementary in two ways. First, rather than stopping at aggregate training or test error, we identify *directional* components of training error relative to a data-dependent geometric object. Second, in the RFNN setting, building on random-feature analyses of diffusion training (George et al., 2025; Bonnaire et al., 2025), we show how limited expressivity due to finite width and incomplete optimization translate into different geometric effects during inference: normal components of error control alignment to the ridge, while tangential components control spreading along it. In this sense, our framework explains not only what bias training creates, but also how that bias appears geometrically during sampling.

Inference-time bias accumulation: metric and geometric viewpoints. Another broad perspective studies how errors accumulate through inference. One version of this literature (e.g., Lee et al., 2022; Chen et al., 2023a; Benton et al., 2024; Wang et al., 2024) characterizes the gap between exact and learned reverse processes through divergence-type quantities such as KL or TV. Such results are useful for quantifying distributional discrepancy, but by themselves they say relatively

little about the geometry of generated samples or how those samples are organized relative to the training data beyond a global metric.

A second version (Chen, 2025; Baptista et al., 2025; Farghly et al., 2025; Li et al., 2025) takes a more geometric viewpoint, often under manifold-type assumptions on the data distribution. Our work is closest in spirit to this line, but differs in several important respects. First, we make the relevant data structure explicit by constructing a time-indexed family of log-density ridge sets directly from the smoothed empirical distribution, rather than assuming an underlying manifold a priori. This differs from Farghly et al. (2025), which motivates log-density smoothing as a useful analytical lens but does not study simulated inference trajectories or provide a data-dependent ridge-manifold description. It is also complementary to Baptista et al. (2025), which analyzes memorization through the reverse dynamics induced by the empirical-loss minimizer using Voronoi geometry, whereas our focus is the complementary non-memorizing regime, where we identify a time-dependent ridge geometry from the smoothed empirical distribution and analyze how inference evolves relative to it. Second, we analyze how inference trajectories evolve relative to this geometry through a reach–align–slide mechanism. This goes beyond concentration near a low-dimensional set by capturing part of the tangential organization of generation relative to nearby data, while remaining intentionally partial: the analysis predicts tangential motion toward nearby data-induced centers rather than fully characterizing the generated configuration inside the tangent space. Third, our setting complements Li et al. (2025), which studies the transition between sampling uniformly from a low-dimensional manifold and sampling a target distribution supported on that manifold. In contrast, we study full denoising diffusion, where the noise level is time-dependent and vanishes as sampling approaches the data, and we explicitly quantify how the learned score or posterior mean estimator drives inference around the geometric object governing generation. Finally, compared with analytical explanations of interpolation bias in stylized settings or under imposed error ansatz (e.g., Chen, 2025), our conclusions are derived under verifiable regularity conditions and connect the resulting geometric bias explicitly to both the dataset and the training parameters.

How our framework combines these perspectives. Viewed together, our framework combines several of the above viewpoints in a single data-driven analysis. We work directly with the empirical data distribution, avoiding any need to assume an underlying smooth population distribution or a fixed manifold known in advance. We then construct a time-dependent ridge family adapted to the diffusion noise level and prove that reverse-time inference evolves relative to this family through a reach–align–slide mechanism. Finally, in the RFNN+GD setting, we decompose the relevant directional training errors into architecture-driven and optimization-driven terms, making explicit how model class, training procedure, and inference geometry interact. Throughout, the analysis is nonasymptotic: the dataset is finite, and in the RFNN example the data dimension, sample size, network width, and training time are all kept finite rather than taken to infinity.

Summary. Relative to nearby work, the main distinction of our paper is therefore not a single isolated technical improvement, but a shift in viewpoint. We study non-memorizing generation in a fully data-dependent setting, identify an explicit time-dependent geometric object from the empirical data, analyze reverse-time inference relative to that object through reach-align-slide, and connect the resulting geometry back to directional components of training error.

B. Denoising Mean Matching Loss

In this section, we introduce the detailed derivation of the posterior mean-matching loss $\mathcal{L}_{\mathrm{MM}}$ and the denoising posterior mean-matching loss $\mathcal{L}_{\mathrm{DMM}}$. According to Tweedie's formula that

$$\nabla \log p_t(x) = -\frac{x}{h_t} + \frac{\mathbb{E}[a_t X_0 | X_t = x]}{h_t} = -\frac{x}{h_t} + \frac{m(t,x)}{h_t},$$

to parametrize the score, it suffices to parametrize the posterior mean m. We denote the parametrization by m_A. Then the posterior mean matching loss $\mathcal{L}_{\mathrm{MM}}$ defined in (4) is equivalent to the score matching loss:

$$\begin{aligned}
\mathcal{L}_{\mathrm{MM}} &= \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\big[\|m_A(t,X_t) - m(t,X_t)\|^2\big] \mathrm{d}t \\
&= \int_\delta^T w(t) \mathbb{E}\big[\| \frac{-X_t + m_A(t,X_t)}{h_t} - \frac{-X_t + m(t,X_t)}{h_t} \|^2\big] \mathrm{d}t \\
&= \int_\delta^T w(t) \mathbb{E}\big[\|s_A(t,X_t) - \nabla \log p_t(X_t)\|^2\big] \mathrm{d}t.
\end{aligned}$$

However, $\mathcal{L}_{\mathrm{MM}}$ can't be evaluated directly using data from X_0. We can apply the same denoising trick as what's done for score matching loss.

$$\begin{aligned}
\mathcal{L}_{\mathrm{MM}} &= \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}\big[\|m_A(t,X_t) - m(t,X_t)\|^2\big]\mathrm{d}t \\
&= \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}\big[\|m_A(t,X_t) - a_tX_0 + a_tX_0 - m(t,X_t)\|^2\big]\mathrm{d}t \\
&= \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}\big[\|m_A(t,X_t) - a_tX_0\|^2\big]\mathrm{d}t + \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}\big[\|m(t,X_t) - a_tX_0\|^2\big]\mathrm{d}t \\
&\quad - 2\int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}\big[\langle m_A(t,X_t) - a_tX_0, m(t,X_t) - a_tX_0\rangle\big]\mathrm{d}t \\
&= \mathcal{L}_{\mathrm{DMM}} + C,
\end{aligned}$$

where the last term in the third identity is canceled due to the definition of m and tower property. The second term in the third identity is a constant independent to the trained parameter A. Therefore, we can train to optimize $\mathcal{L}_{\mathrm{DMM}}$ directly. The DMM loss evolves two expectations and one integral:

$$\mathcal{L}_{\mathrm{DMM}} = \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}_{X_0}\mathbb{E}_z\big[\| - a_tX_0 + m_A(t, a_tX_0 + \sqrt{h_t}z)\|^2\big]\mathrm{d}t.$$

Under data assumption 3.1 that $p = \frac{1}{n}\sum_{i=1}^n \delta_{x_0^{(i)}}$, the expectation $\mathbb{E}_{X_0}$ can be exactly evaluated through empirical average over all training data, i.e.,

$$\mathcal{L}_{\mathrm{DMM}} = \frac{1}{n}\sum_{i=1}^n \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}_z\big[\| - a_tx_0^{(i)} + m_A(t, a_tx_0^{(i)} + \sqrt{h_t}z)\|^2\big]\mathrm{d}t. \tag{10}$$

For the convenience of analysis, we focus on analyzing the loss defined in (10), which corresponds to exact evaluations for $\mathbb{E}_z$ and integral in t.

In practice, the loss in (10) is used after further numerical approximations for $\mathbb{E}_z$ and integral in t. The practical DMM loss is given by

$$\mathcal{L}_{\mathrm{DMM}}^{m,N} = \frac{1}{nm}\sum_{i,j=1}^{n,m}\sum_{k=1}^{N}\frac{t_k - t_{k-1}}{h_{t_k}^2}\| - a_{t_k}x_0^{(i)} + m_A(t_k, a_{t_k}x_0^{(i)} + \sqrt{h_{t_k}}z^{(i,j)})\|^2$$

where $\{z^{(i,j)}\}_{1\le i\le n, 1\le j\le m}$ is a sequence of i.i.d. standard Gaussian vectors in $\mathbb{R}^d$ and $\delta = t_0 < t_1 < \cdots < t_N = T$ are the time grids for numerical integration on $[\delta, T]$.

C. Data-Independent Properties of the Log-density Ridge Sets

In this section, we introduce properties of the log-density ridges that are independent to our data assumptions. We summarize them in the following Proposition.

Proposition C.1. *Under Assumption 3.3, for any $\rho_t \in (0, r_t]$ and the tube neighborhood $\mathcal{T}_t(\rho_t)$ given below*

$$\mathcal{T}_t(\rho_t) \coloneqq \{x \in \mathbb{R}^d | \mathrm{dist}(x, \mathcal{R}_t) \le \rho_t\}, \tag{11}$$

the nearest-point projection $\Pi_t : \mathcal{T}_t(\rho_t) \to \mathcal{R}_t$ is well-defined and we have

(1) Π_t is C^1 on $\mathcal{T}_t(\rho_t)$;

(2) $\forall x \in \mathcal{T}_t(\rho_t)$, $n_t(x) \coloneqq x - \Pi_t(x)$ is in the normal space at $\Pi_t(x)$, i.e, $n_t(x) \in N_{\Pi_t(x)}(\mathcal{R}_t)$;

(3) $\sup_{x\in\mathcal{T}_t(\rho_t)} \|\nabla\Pi_t(x)\| \le \frac{1}{1-\rho_t/r_t}$;

(4) the motion of Π_t *is bounded: for any* $z \in \mathcal{R}_t$*, there exists a velocity field* $v_t \in N_z(\mathcal{R}_t)$ *s.t.* $\sup_{x\in\mathcal{T}_t(\rho_t)} \|\partial_t \Pi_t\| \le V_t + \frac{\rho_t}{1-\rho_t/r_t} W_t$ *where*

$$V_t = \sup_{z\in\mathcal{R}_t} \|v_t(z)\|, \quad W_t = \sup_{z\in\mathcal{R}_t, \|u\|=1} \|P^{\|}(z)(\nabla v_t(z)u)\|^2.$$

Proof of Proposition C.1. That Π_t is well-defined on $\mathcal{T}_t(\rho_t)$ directly follows from Assumption 3.3. The C^1 smoothness of Π_t in space follows from Leobacher & Steinicke (2021, Theorem 2). Property (2) follows from the optimality of the nearest-point: $z = \Pi_t(x)$ minimizes $\|x - z'\|^2$ over $z' \in \mathcal{R}_t$. Therefore, differentiating $z' \mapsto \|x - z'\|^2$ along any tangent direction $u \in T_z(\mathcal{R}_t)$ yields

$$D\big(\|x - z'\|^2\big)[u]|_{z'=z} = -2\langle x - z, u\rangle = -2\langle n_t(x), u\rangle = 0.$$

Therefore, $n_t(x) \in N_z(\mathcal{R}_t)$.

To prove (3), we first apply the explicit expression of $\nabla\Pi_t(x)$ in Leobacher & Steinicke (2021, Theorem C):

$$\nabla\Pi_t(x) = \big(id_{T_{\Pi_t(x)}(\mathcal{R}_t)} - \|x - \Pi_t(x)\| S_{t,\Pi_t(x),v}\big)^{-1} P^{\|}(\Pi_t(x)), \quad \theta = \frac{x - \Pi_t(x)}{\|x - \Pi_t(x)\|},$$

where $S_{t,\Pi_t(x),\theta}$ is the shape operator in the normal direction θ. According to Lemma C.3 and the linearity of $S_{t,\Pi_t(x),\theta}$ in θ, we have

$$\begin{aligned}\|\nabla\Pi_t(x)\| &\le \|\big(id_{T_{\Pi_t(x)}(\mathcal{R}_t)} - \|x - \Pi_t(x)\| S_{t,\Pi_t(x),v}\big)^{-1} P^{\|}(\Pi_t(x))\| \\ &\le \|\big(id_{T_{\Pi_t(x)}(\mathcal{R}_t)} - S_{t,\Pi_t(x),\|x-\Pi_t(x)\|v}\big)^{-1}\| \le \frac{1}{1 - \|x - \Pi_t(x)\|/r_t} \le \frac{1}{1 - \rho_t/r_t}.\end{aligned}$$

Last, to prove (4), we first apply Lemma C.4 to show existence of the normal velocity field v_t. Then the estimation of $\|\partial_t\Pi_t\|$ follows from the definition of V_t and Lemma C.5. □

Definition C.2 (Shape Operator)**.** Let $\mathrm{II}_{t,z}$ be the second fundamental form of $\mathcal{R}_t$ at $z \in \mathcal{R}_t$. For $\theta \in N_z(\mathcal{R}_t)$, the shape operator in the direction of θ is defined as $S_{t,z,\theta} : T_z(\mathcal{R}_t) \to T_z(\mathcal{R}_t)$ is then defined as

$$\langle S_{t,z,\theta}u, v\rangle = \langle \mathrm{II}_{t,z}(u, v), \theta\rangle, \quad \forall u, v \in T_z(\mathcal{R}_t).$$

Lemma C.3. *Under Assumption 3.3, for any* $z \in \mathcal{R}_t$ *and* $\theta \in N_z(\mathcal{R}_t)$*, we have*

(1) the shape operator is bounded: $\|S_{t,z,\theta}\| \le \|\theta\|/r_t$*;*

(2) if $\|\theta\| \le \rho_t < r_t$*, the operator* $L_{t,z,\theta} := id_{T_z(\mathcal{R}_t)} - S_{t,z,\theta}$ *is invertible and*

$$\|L^{-1}_{t,z,\theta}\| \le \frac{1}{1 - \|\theta\|/r_t}.$$

Proof of Lemma C.3. According to Niyogi et al. (2008, Proposition 6.1), under Assumption 3.3, $\|\mathrm{II}_{t,z}\| \le 1/r_t$ for all t. Therefore, $\|S_{t,z,\theta}\| \le \|\theta\|/r_t$ and hence $\|L^{-1}_{t,z,\theta}\| = \|(id_{T_z(\mathcal{R}_t)} - S_{t,z,\theta})^{-1}\| \le \frac{1}{1-\|\theta\|/r_t}$. □

Lemma C.4. *Fix* $t \in [\delta, T]$*. Let* $U \subset \mathbb{R}^{d^*}$ *be open and* $\tilde{\Phi}_t : U \to \mathbb{R}^d$ *be a* C^1*-family of* C^2 *embeddings such that* $\tilde{\Phi}_t(U) \subset \mathcal{R}_t$ *and* $\tilde{\Phi}_t$ *is a local parametrization of* $\mathcal{R}_t$ *along a given* C^1*-curve* $z(t) \in \mathcal{R}_t$*. Assume that* $z(t) = \tilde{\Phi}_t(\tilde{u}(t))$ *for some* C^1*-curve* $\tilde{u}(t) \in U$*. Then there exists a* C^1*-family of local diffeomorphisms* $\psi_t : U' \to U$ *Then there exists a family of diffeomorphism* $\psi_t : U' \to U$ *such that*

$$\Phi_t := \tilde{\Phi}_t \circ \psi_t : U' \to \mathcal{R}_t, \quad \partial_t\Phi_t(u) \perp T_{\Phi_t(u)}(\mathcal{R}_t),\ \forall (t, u) \in [\delta, T] \times U'.$$

Consequently, the velocity field $v_t(z) := \partial_t\Phi_t(u)$ *for* $z = \Phi_t(u)$ *is a well-defined* C^0 *normal velocity field on* $\Phi(U')$*, i.e.,* $v_t(z) \in N_z(\mathcal{R}_t)$*. Moreover,* v_t *is intrinsic in the sense that it equals to the normal component of* $\partial_t\tilde{\Phi}_t$*:* $v_t(\Phi_t(u)) = P_{N_{\Phi_t(u)}}(\partial_t\tilde{\Phi}_t(\psi_t(u)))$*, and is therefore independent of the tangent reparametrization of the chart.*

[2]For any $z \in \mathcal{R}_t$, we use $P^{\|}(z)$ (or $P^{\perp}(z)$) to represent the orthogonal projection from $\mathbb{R}^d$ to $T_z(\mathcal{R}_t)$ (or $N_z(\mathcal{R}_t)$).

Proof of Lemma C.4. Define $\tilde{v}_t(u) \coloneqq \partial_t \tilde{\Phi}_t(u)$. Since $\tilde{\Phi}_t$ is an embedding, $\nabla \tilde{\Phi}_t(u) : \mathbb{R}^{d^*} \to T_{\tilde{\Phi}_t(u)}(\mathcal{R}_t)$ is a linear isomorphism for each (t, u). Let $P^{\parallel}(\tilde{\Phi}_t(u))$ denote orthogonal projection onto $T_{\tilde{\Phi}_t(u)}(\mathcal{R}_t)$. Define a time-dependent vector field a_t on U by

$$\nabla \tilde{\Phi}_t(u) a_t(u) = -P^{\parallel}(\tilde{\Phi}_t(u))\tilde{v}_t(u) \in T_{\tilde{\Phi}_t(u)}(\mathcal{R}_t). \tag{12}$$

Since $\nabla \tilde{\Phi}_t(u)$ is invertible on $T_{\tilde{\Phi}_t(u)}(\mathcal{R}_t)$, $a_t(u)$ in (12) is uniquely defined. Furthermore, under our assumptions, $(t, u) \mapsto \nabla \tilde{\Phi}_t(u)$ and $(t, u) \mapsto P^{\parallel}(\tilde{\Phi}_t(u))\tilde{v}_t(u)$ are continuous in t and smooth in u, hence $(t, u) \mapsto a_t(u)$ is also continuous in t and smooth in u.

Pick an open set $U' \subset U$ that contains $\tilde{u}(t)$ for all $t \in [\delta, T]$. Consider the ODE

$$\partial_t \psi_t(u) = a_t(\psi_t(u)), \quad \psi_\delta(u) = u \in U'. \tag{13}$$

Due to the regularity of $a_t(\cdot)$, there exists a unique solution ψ_t for all $t \in [\delta, T]$ and ψ_t is a diffeomorphism for each t.

Now apply chain rule and we get

$$\begin{aligned}
\partial_t \Phi_t(u) &= \partial_t(\tilde{\Phi}_t \circ \psi_t)(u) = \tilde{v}_t(\psi_t(u)) + \nabla \tilde{\Phi}_t(\psi_t(u))\partial_t \psi_t(u) \\
&= \tilde{v}_t(\psi_t(u)) + \nabla \tilde{\Phi}_t(\psi_t(u)) a_t(\psi_t(u)) = \tilde{v}_t(\psi_t(u)) - P^{\parallel}(\tilde{\Phi}_t(\psi_t(u)))\tilde{v}_t(\psi_t(u)) \\
&= P^{\perp}(\tilde{\Phi}_t(\psi_t(u)))\tilde{v}_t(\psi_t(u)),
\end{aligned}$$

where the second last identity follows from (12) and $P^{\perp}(\tilde{\Phi}_t(\psi_t(u))) = I - P^{\parallel}(\tilde{\Phi}_t(\psi_t(u)))$ is the normal projection. Therefore, $\partial \Phi_t(u) \perp T_{\tilde{\Phi}_t(\psi_t(u))}(\mathcal{R}_t)$. Last, $v_t(z) \coloneqq \partial \Phi_t(u) \in N_{\tilde{\Phi}_t(\psi_t(u))}(\mathcal{R}_t)$ with $z = \Phi_t(u)$ is well-defined on $\Phi_t(U')$ and we can check that $v_t(\Phi_t(u))$ is exactly the normal component of $\partial_t \tilde{\Phi}_t(\psi_t(u))$, hence independent to the tangent reparametrization. □

Lemma C.5. *For $t \in [\delta, T]$, let $x \in \mathcal{T}_t(\rho_t)$ and $z(t) \coloneqq \Pi_t(x)$. Under conditions in Lemma C.4 and Assumption 3.3, there exists a reparametrization chart Φ_t in normal gauge and a C^1 curve $u(t) \in U \subset \mathbb{R}^{d^*}$ such that $z(t) = \Phi_t(u(t))$ and*

$$\partial_t \Pi_t(x) = v_t(z(t)) + \tau_t, \quad v_t(z(t)) \in N_{z(t)}(\mathcal{R}_t) \text{ and } \tau_t \coloneqq \nabla_u \Phi_t(u(t))\partial_t u(t) \in T_{z(t)}(\mathcal{R}_t).$$

Furthermore, $\|\tau_t\| \le \frac{\rho_t}{1-\rho_t/r_t} W_t$.

Proof of Lemma C.5. The existence of Φ_t follows from Lemma C.4. Next, differentiate $z(t) = \Phi_t(u(t))$ and we get

$$\partial_t z(t) = \partial_t \Phi_t(u(t)) + \nabla_u \Phi_t(u(t))\partial_t u(t) = v_t(z) + \tau_t.$$

To prove the bound for $\|\tau_t\|$, we consider the local tangent frame $\{E_i(t)\}_{i=1}^{d^*}$ along $z(t)$ induced by the chart $u \mapsto \Phi_t(u)$. Since $n_t(x) = x - z(t) \perp T_{z(t)}(\mathcal{R}_t)$, we have $\langle n_t(x), E_i(t)\rangle = 0$ for all $1 \le i \le d^*$. Differentiate wrt t on both sides and we get

$$0 = -\langle \partial_t z(t), E_i(t)\rangle + \langle n_t(x), \partial_t E_i(t)\rangle.$$

Decompose $\partial_t z(t) = v_t(z(t)) + \tau_t$ and use the fact that $v_t(z(t)) \perp T_{z(t)}(\mathcal{R}_t)$, and we get

$$\langle \tau_t, E_i(t)\rangle = \langle n_t(x), \partial_t E_i(t)\rangle. \tag{14}$$

Using the local frame we can compute $\partial_t E_i(t)$ as follows. Since $E_i(t) = \partial_{u_i} \Phi_t(u(t))$, we have

$$\begin{aligned}
\partial_t E_i(t) &= \partial_{u_i} \partial_t \Phi_t(u(t)) + \sum_{j=1}^{d^*} \partial^2_{u_i u_j} \Phi_t(u(t))\partial_j u(t) \\
&= \partial_{u_i}(v_t \circ \Phi_t)(u(t)) + \sum_{j=1}^{d^*} P^{\perp}(z)(\partial^2_{u_i u_j} \Phi_t(u(t)))\partial_t u_j(t) + \sum_{j=1}^{d^*} P^{\parallel}(z)(\partial^2_{u_i u_j} \Phi_t(u(t)))\partial_t u_j(t) \\
&= \nabla v_t(z) E_i(t) + \sum_{j=1}^{d^*} \mathrm{II}_{t,z}(E_i(t), E_j(t))\partial_t u_j(t) + \sum_{j=1}^{d^*} P^{\parallel}(z)(\partial^2_{u_i u_j} \Phi_t(u(t)))\partial_t u_j(t).
\end{aligned}$$

Since $n_t(x) \in N_{z(t)}(\mathcal{R}_t)$, we have

$$\begin{aligned}\langle n_t(x), \partial_t E_i(t)\rangle &= \langle n_t(x), \nabla v_t(z(t))E_i(t)\rangle + \langle n_t(x), \mathrm{II}_{t,z}(E_i(t), \sum_j \partial_t u_j(t)E_j(t))\rangle \\ &= \langle n_t(x), \nabla v_t(z(t))E_i(t)\rangle + \langle S_{t,z,n_t(x)}\tau_t, E_i(t)\rangle,\end{aligned}$$

where the last identity follows from the definition of shape operator and $\tau_t = \sum_j \partial_t u_j(t)E_j(t)$. Plug the above equation into (14) and we get that restricted to the tangent space $\mathcal{T}_{z(t)}(\mathcal{R}_t)$,

$$(I - S_{t,z,n_t(x)})\tau_t = P^{\parallel}(z(t))(\nabla v_t(z(t)))^{\intercal} n_t(x).$$

Therefore, according to Lemma C.3, the definition of W_t and the fact that $x \in \mathcal{T}_t(\rho_t)$,

$$\|\tau_t\| \le \|L^{-1}_{t,z,n_t(x)}\|\|\nabla v_t(z(t))\|\|n_t(x)\| \le \frac{\rho_t}{1-\rho_t/r_t} W_t.$$

□

D. Data dependent Ridge Motion Estimations

In this section, we provided some data-dependent estimations for quantities related to dynamical properties of the log-density ridges.

Proposition D.1. *Under Assumption 3.1, the log-density ridge sets satisfy Assumption 3.3. Furthermore, when $t \to \delta^+ \ll 1$, we have the following order estimations:*

$$r_t = \Omega(\beta_t h_t^3 \theta_t^{-1} R^{-3}), \quad V_t = \mathcal{O}(\beta_t^{-1} h_t^{-1} R), \quad W_t = \mathcal{O}(\beta_t^{-1} h_t^{-1} R r_t^{-1}),$$

where θ_t is arbitrarily with order $\theta_t = \exp(-o(h_t^{-1}))$.

Remark D.2 (Order estimation of ridge motion). Combining Propositions C.1 and D.1, picking $\rho_t = \Theta(r_t)$ and $\beta_t = \Theta(1/h_t)$, we proved Proposition 3.4-(3): $\sup_{x\in\mathcal{T}_t(\rho_t)} \|\partial_t \Pi_t(x)\| = \mathcal{O}(R)$.

To study the relation between the data Assumption 3.1 and properties of the data-dependent manifold $\mathcal{R}_t$. The key is to estimate the derivatives of the score $\nabla \log p_{T-t}(\cdot)$ in different regions dominated by centers $\{m^{(i)}_{T-t} \coloneqq a_{T-t}x_0^{(i)}\}_{i=1}^n$. For each $i \in [n]$ and $|\theta_s| < 1$ for any $\zeta > 0$ when $s \to 0^+$, define the center-i dominate region

$$\mathcal{B}^{(i)}_{T-t}(\theta_{T-t}) \coloneqq \{x \in \mathbb{R}^d \mid \mathrm{Softmax}(-\frac{\|x - a_{T-t}x_0\|^2}{2h_{T-t}})_i \ge 1 - \theta_{T-t}\}. \tag{15}$$

Next, we introduce a sufficient condition for $x \in \mathcal{B}^{(i)}_{T-t}(\theta_{T-t})$.

Lemma D.3. *For any $x \in \mathbb{R}^d$ and $\theta_{T-t} \in (0, \frac{1}{2})$, if*

$$\|x - m^{(j)}_{T-t}\|^2 - \|x - m^{(i)}_{T-t}\|^2 \ge 2h_{T-t}\log\big(\frac{(1-\theta_{T-t})(n-1)}{\theta_{T-t}}\big), \quad \forall j \ne i, \tag{16}$$

then $x \in \mathcal{B}^{(i)}_{T-t}(\theta_{T-t})$.

Remark D.4. As $t \to T^-$, if $h_{T-t} = o(1)$ and $\theta_{T-t} = \exp(-o(h_{T-t}^{-1}))$, then RHS of (16) is of order $o(1)$. Therefore, Y_t (or $\tilde{Y}_t$) satisfies the (16) with probability 1 as $t \to T^-$. As a consequence of Lemma D.3, $Y_t \in \mathcal{B}^{(i)}_{T-t}(\theta_{T-t})$ (or $\tilde{Y}_t \in \mathcal{B}^{(i)}_{T-t}(\theta_{T-t})$) with probability 1 as $t \to T^-$.

Proof of Lemma D.3. Under (16), we have that for all $j \ne i$,

$$\frac{\mathrm{Softmax}(-\frac{\|x-m_{T-t}\|^2}{2h_{T-t}})_j}{\mathrm{Softmax}(-\frac{\|x-m_{T-t}\|^2}{2h_{T-t}})_i} = \exp\big(-\frac{\|x - m^{(j)}_{T-t}\|^2 - \|x - m^{(i)}_{T-t}\|^2}{2h_{T-t}}\big) \le \frac{\theta_{T-t}}{(1-\theta_{T-t})(n-1)}.$$

Therefore,

$$\begin{aligned}&1-\mathrm{Softmax}(-\frac{\|x-m_{T-t}\|^2}{2h_{T-t}})_i\\&=\sum_{j\neq i}\mathrm{Softmax}(-\frac{\|x-m_{T-t}\|^2}{2h_{T-t}})_j\leq\frac{\theta_{T-t}}{(1-\theta_{T-t})}\mathrm{Softmax}(-\frac{\|x-m_{T-t}\|^2}{2h_{T-t}})_i.\end{aligned}$$

The statement follows from the definition of $\mathcal{B}^{(i)}_{T-t}(\eta_{T-t})$. □

Now we provide estimates of derivatives of the score on the center-dominate regions.

Lemma D.5. *For any $t\in[\delta,T]$, we have*

$$\sup_{x\in\cup_{i=1}^n\mathcal{B}_t^{(i)}(\theta_t)}\|\nabla m(t,x)\|\leq\frac{20a_t^2\theta_tR^2}{h_t},\qquad\sup_{x\in\cup_{i=1}^n\mathcal{B}_t^{(i)}(\theta_t)}\|\nabla^3\log p_t(x)\|\leq\frac{80a_t^3\theta_tR^3}{h_t^3},$$

where $\mathcal{B}_t^{(i)}(\theta_t)$ is defined in (15) with any $\theta_t=\exp(-o(h_t^{-1}))$.

Remark D.6 (Choice of β_t). Lemma D.5 also validates the choice of ridge threshold $\beta_t=\Theta(1/h_t)$ when $t\to0^+$. According to Lemma D.7, $\nabla^2\log p_t(x)=-\frac{1}{h_t}I_d+\frac{1}{h_t^2}\Sigma(t,x)$. According to estimations in Lemma D.5, each eigenvalue $\lambda_j(t,x)$ of $\nabla^2\log p_t(x)$ satisfies

$$\lambda_j(t,x)\leq-\frac{1}{h_t}+\frac{20a_t^2\theta_tR^2}{h_t^2}\leq-\frac{c}{h_t},\quad\frac{1}{2}<c<1,$$

under some choice of $\theta_t=\exp(-o(h_t^{-1}))$. Therefore, as $t\to0^+$, the choice of $\beta_t=c/h_t$ makes the second condition in the log-density ridge definition automatically satisfied.

Proof of Lemma D.5. According to Lemma D.7, for all $x\in\mathcal{B}_t^{(i)}(\theta_t)$

$$\|\nabla m(t,x)\|=\frac{1}{h_t}\|\Sigma(t,x)\|\leq\frac{1}{h_t}\sum_{j=1}^n\mathrm{Softmax}(-\frac{\|x-a_tx_0\|^2}{2h_t})_j\|a_tx_0^{(j)}-m(t,x)\|^2.$$

Notice that for $j=i$,

$$\begin{aligned}\|a_tx_0^{(i)}-m(t,x)\|&=\|a_tx_0^{(i)}-\sum_{j'=1}^n\mathrm{Softmax}(-\frac{\|x-a_tx_0\|^2}{2h_t})_{j'}a_tx_0^{(j')}\|\\&\leq\sum_{j'\neq i}\mathrm{Softmax}(-\frac{\|x-a_tx_0\|^2}{2h_t})_{j'}\|a_tx_0^{(j')}-a_tx_0^{(i)}\|\leq2a_t\theta_tR.\end{aligned}$$

For $j\neq i$, we have

$$\|a_tx_0^{(j)}-m(t,x)\|\leq\|a_tx_0^{(i)}-m(t,x)\|+\|a_tx_0^{(i)}-a_tx_0^{(j)}\|\leq2a_t(\theta_t+1)R\leq4a_tR.$$

Combining the above estimations and we get

$$\begin{aligned}&\|\nabla m(t,x)\|\\&\leq\frac{1}{h_t}\mathrm{Softmax}(-\frac{\|x-a_tx_0\|^2}{2h_t})_i(2a_t\theta_tR)^2+\frac{1}{h_t}(\sum_{j\neq i}\mathrm{Softmax}(-\frac{\|x-a_tx_0\|^2}{2h_t})_j)(4a_tR)^2\\&\leq\frac{20a_t^2\theta_tR^2}{h_t}.\end{aligned}$$

Following the same approach, we can bound $\|\nabla^3 \log p_t(x)\|$ for $x \in \mathcal{B}_t^{(i)}(\theta_t)$:

$$\begin{aligned}
&\|\nabla^3 \log p_t(x)\| \\
\leq& \frac{1}{h_t^3}\sum_{j=1}^{n} \mathrm{Softmax}(-\frac{\|x-a_t x_0\|^2}{2h_t})_j \|a_t x_0^{(j)} - m(t,x)\|^3 \\
\leq& \frac{1}{h_t^3}(\mathrm{Softmax}(-\frac{\|x-a_t x_0\|^2}{2h_t})_i)(2a_t\theta_t R)^3 + \frac{1}{h_t^3}(\sum_{j\neq i}\mathrm{Softmax}(-\frac{\|x-a_t x_0\|^2}{2h_t})_j)(4a_t R)^3 \\
\leq& \frac{80 a_t^3 \theta_t R^3}{h_t^3}.
\end{aligned}$$

□

Now we introduce the proof of Proposition D.1.

Proof of Proposition D.1. According to Lemma D.8, we have

$$\sup_{z\in\mathcal{R}_t} \|\mathrm{II}_{t,z}\| \leq \frac{1}{\beta_t}\sup_{z\in\mathcal{R}_t}\|\nabla^3 \log p_t(z)\|.$$

Since reach is at least the reciprocal of maximal curvature, we derive that

$$r_t \gtrsim \beta_t(\sup_{z\in\mathcal{R}_t}\|\nabla^3 \log p_t(z)\|)^{-1}.$$

As $t\to 0^+$, according to Lemma D.3 and definition of $\mathcal{B}_t^{(i)}(\theta_t)$, $x \in \cup_{i=1}^n \mathcal{B}_t^{(i)}(\theta_t)$ with probability 1 and $\overline{\cup_{i=1}^n \mathcal{B}_t^{(i)}(\theta_t))} \to \mathbb{R}^d$. Therefore, as $t\to 0^+$, we have

$$r_t \gtrsim \beta_t(\sup_{z\in\mathcal{R}_t}\|\nabla^3 \log p_t(z)\|)^{-1} \gtrsim \beta_t(\sup_{x\in\overline{\cup_{i=1}^n \mathcal{B}_t^{(i)}(\theta_t)}}\|\nabla^3 \log p_t(x)\|)^{-1} = \Omega(\frac{\beta_t h_t^3}{\theta_t R^3}),$$

for any $\theta_t = \exp(-o(h_t^{-1}))$ and the last estimation follows from Lemma D.5.

To bound V_t, recall that $V_t = \sup_{z\in\mathcal{R}_t}\|v_t(z)\|$ with $v_t(z)$ being the velocity field induced by the normal gauge in Lemma C.4.

According to Lemma D.9,

$$\begin{aligned}
v_t(z) &= -(E_t(z)\nabla^2 \log p_t(z) E_t(z))^{-1} E_t(z)^\intercal \partial_t \nabla \log p_t(z) \\
&= -(E_t(z)\nabla^2 \log p_t(z) E_t(z))^{-1}\frac{1}{h_t} E_t(z)^\intercal \partial_t m(t,z),
\end{aligned}$$

where the last identity follows from Lemma D.10. Therefore,

$$\begin{aligned}
V_t &\leq \sup_{z\in\mathcal{R}_t}\|v_t(z)\| \\
&\leq \frac{1}{h_t}\sup_{z\in\mathcal{R}_t}\|(E_t(z)\nabla^2 \log p_t(z)E_t(z))^{-1}\|\|E_t(z)^\intercal \partial_t m(t,z)\| \\
&\leq \frac{1}{\beta_t h_t}\sup_{z\in\mathcal{R}_t}\|E_t(z)^\intercal \partial_t m(t,z)\|,
\end{aligned}$$

where the last inequality follows form the estimate in the proof of Lemma D.9. Hence according to Lemma D.11, $V_t = \mathcal{O}(\frac{R}{h_t\beta_t})$.

Last, to bound W_t, recall that $W_t = \sup_{z\in\mathcal{R}_t,\|u\|=1}\|P^\parallel(z)(\nabla v_t(z)u)\|$. Notice that $v_t(z) \in N_z(\mathcal{R}_t)$. Hence $P^\perp(z)v_t(z) = v_t(z)$. Differentiate both sides of the equation at z along direction u, we have

$$\begin{aligned}
&(\nabla_u P^\perp(z))v_t(z) + P^\perp(z)\nabla_u v_t(z) = \nabla_u v_t(z) \\
\Longrightarrow\ &(\nabla_u P^\perp(z))v_t(z) = (I - P^\perp(z))\nabla_u v_t(z) = P^\parallel(z)\nabla_u v_t(z).
\end{aligned}$$

Therefore, we immediately obtain

$$W_t \le \big(\sup_{z\in\mathcal{R}_t, \|u\|=1} \|\nabla_u P^\perp(z)\| \big) V_t$$

Next, we bound $\|\nabla_u P^\parallel(z)\|$. Notice that $P^\parallel(z) + P^\perp(z) = I_d$, hence $\|\nabla_u P^\parallel(z)\| = \|\nabla_u P^\perp(z)\|$. Now we start from $P^\perp(z)^2 = P^\perp(z)$, taking the directional derivative on both side, left multiplying $P^\parallel(z)$ and right multiplying $P^\parallel(z)$,

$$P^\parallel(z)(\nabla_u P^\parallel(z))P^\parallel(z) + P^\parallel(z)(\nabla_u P^\parallel(z))P^\parallel(z) = P^\parallel(z)\nabla_u P^\parallel(z)P^\parallel(z),$$

Hence $P^\parallel(z)\nabla_u P^\parallel(z)P^\parallel(z) = 0$. Similarly, we start from $P^\perp(z)^2 = P^\perp(z)$, taking the directional derivative on both side, left multiplying $P^\perp(z)$ and right multiplying $P^\perp(z)$, and get

$$P^\perp(z)(\nabla_u P^\parallel(z))P^\parallel(z)P^\perp(z) + P^\perp(z)P^\parallel(z)(\nabla_u P^\parallel(z))P^\perp(z) = P^\perp(z)(\nabla_u P^\parallel(z))P^\perp(z).$$

Since $P^\parallel(z)P^\perp(z) = P^\perp(z)P^\parallel(z)$, we can express $\nabla_u P^\parallel(z)$ on $T_z(\mathcal{R}_t) \oplus N_z(\mathcal{R}_t)$ as

$$\nabla_u P^\parallel(z) = \begin{pmatrix} 0 & B^\intercal \\ B & 0 \end{pmatrix},$$

with $B = P^\perp(z)(\nabla_u P^\parallel(z))P^\parallel(z) : T_z(\mathcal{R}_t) \to N_z(\mathcal{R}_t)$. Therefore,

$$\|\nabla_u P^\perp(z)\| = \|\nabla_u P^\parallel(z)\| = \|B\| = \|\mathrm{II}_{t,z}(u,\cdot)\|_{T_z(\mathcal{R}_t)\to N_z(\mathcal{R}_t)} \lesssim \frac{1}{r_t}.$$

where the last inequality follows from Aamari et al. (2019, Theorem 3.4). Therefore, we have

$$W_t \lesssim V_t / r_t.$$

Hence we proved Proposition D.1-(3). □

Lemma D.7 (Explicit formulas)**.** *Under Assumption 3.1, we have that for all $t \in [\delta, T]$,*

$$\nabla \log p_t(x) = -\frac{x}{h_t} + \frac{m(t,x)}{h_t}, \quad \nabla^2 \log p_t(x) = -\frac{I_d}{h_t} + \frac{\Sigma(t,x)}{h_t^2},$$
$$\nabla^3 \log p_t(x) = \frac{\mathbb{E}[(U(t,x) - m(t,x))^{\otimes 3}]}{h_t^3},$$

where $U(t,x) \in \mathbb{R}^d$ is a random vector taking values $\{a_t x_0^{(i)}\}_{i=1}^n$ with probabilities $\{\mathrm{Softmax}(-\frac{\|x - a_t x_0\|^2}{2h_t})_i\}_{i=1}^n$ and

$$m(t,x) = \mathbb{E}[U(t,x)], \quad \Sigma(t,x) = \mathrm{Cov}(U(t,x)),$$

with $x_0 = (x_0^{(1)}, x_0^{(2)}, \cdots, x_0^{(n)})$. Furthermore, $\nabla m(t,x) = \Sigma(t,x)/h_t$.

Lemma D.8 (Second Fundamental Form Bound)**.** *For any $t \in [\delta, T]$ and $z \in \mathcal{R}_t$, we have*

$$\|\mathrm{II}_{t,z}\| \le \frac{1}{\beta_t} \|\nabla^3 \log p_t(z)\|.$$

Proof of Lemma D.8. For any $z \in \mathcal{R}_t$, we consider $T \in \mathbb{R}^{d\times d^*}$ and $N \in \mathbb{R}^{d\times(d-d^*)}$ to be the orthonormal basis spanning the spaces $T_z(\mathcal{R}_t)$ and $N_z(\mathcal{R}_t)$ respectively. We use coordinates $(u,v) \in \mathbb{R}^{d^*} \times \mathbb{R}^{d-d^*}$ via $x(u,v) = z + Tu + Nv$. We can define the function

$$F(u,v) \coloneqq N^\intercal \nabla \log p_t(x(u,v)) \in \mathbb{R}^{d-d^*}.$$

Since $z \in \mathbb{R}_t$, the definition of ridge set $\mathcal{R}_t$ includes the normal component of the score $\nabla \log p_t(x)$ is zero, i.e., $F(0,0) = 0$. The partial derivatives along the normal direction is

$$\partial_v F(u,v) = N^\intercal \nabla^2 \log p_t(x(u,v)) N, \quad \partial_v F(0,0) = N^\intercal \nabla^2 \log p_t(z) N.$$

Since $\nabla^2 \log p_t$ is invertible, we have

$$\|(\partial_v F(0,0))^{-1}\| = \|(N^\intercal \nabla^2 \log p_t(z) N)^{-1}\| \leq \frac{1}{\beta_t},$$

where the inequality follows from the second condition in the definition of $\mathcal{R}_t$. Therefore, apply the implicit function theorem and we get: there exists a neighborhood $0 \in U \subset \mathbb{R}^{d^*}$ and a C^2 map $\phi : U \to \mathbb{R}^{d-d^*}$ with $\phi(0) = 0$ such that the local solution set to $F(u,v) = 0$ is the set $\{(u,v) \mid v = \phi(u)\}$. Therefore, a local parametrization of the manifold is

$$\gamma(u) = z + Tu + N\phi(u).$$

And differentiating $F(u, \phi(u)) = 0$ at $u = 0$ implies $\nabla\phi(0) = 0$. Now differentiate $F(u, \phi(u)) = 0$ twice and evaluate at $u = 0$ and use the fact that $\nabla\phi(0) = 0$, we get

$$\partial_{uu} F(0,0) + \partial_v F(0,0) \nabla^2 \phi(0) = 0.$$

Since $F(u,v) = N^\intercal \nabla \log p_t(x(u,v))$, we have that for all $\xi \in \mathbb{R}^{d^*}$,

$$\partial_{uu} F(0,0)[\xi,\xi] = N^\intercal (\nabla^3 \log p_t(z))[T\xi, T\xi] \implies \|\partial_{uu} F(0,0)\| \leq \|\nabla^3 \log p_t(z)\|.$$

Therefore, we have

$$\|\nabla^2\phi(0)\| = \| - (\partial_v F(0,0))^{-1} \partial_{uu} F(0,0)\| \leq \| - (\partial_v F(0,0))^{-1}\| \|\partial_{uu} F(0,0)\| \leq \frac{1}{\beta_t} \|\nabla^3 \log p_t(z)\|.$$

Last, along the local parametrization $\gamma(u) = z + Tu + N\phi(u)$, we have

$$\partial^2_{ij} \gamma(0) = N \partial^2_{ij} \phi(0).$$

Therefore, apply the definition of II and we get

$$\mathrm{II}_{t,z}(\nabla\gamma(0)\xi, \nabla\gamma(0)\theta) = P^\perp(z) \nabla^2 \gamma(0)[\xi,\theta] = N \nabla^2 \phi(0)[\xi,\theta].$$

Since N is isotropic (orthonormal), we prove the Lemma. □

Lemma D.9 (Normal velocity field expression)**.** *For all fixed $t \in [\delta, T]$ and $z \in \mathcal{R}_t$, let $E_t \in \mathbb{R}^{d \times (d-d^*)}$ be the normal eigen-matrix in the definition of $\mathcal{R}_t$. The normal velocity field in Lemma C.4 can be expressed as*

$$v_t(z) = -(E_t(z) \nabla^2 \log p_t(z) E_t(z))^{-1} E_t(z) \partial_t \nabla \log p_t(z) \in N_z(\mathcal{R}_t). \tag{17}$$

Proof of Lemma D.9. Fix $t_0 \in (\delta, T)$ and $z_0 \in \mathcal{R}_{t_0}$, consider the normal gauge parametrization near (t_0, z_0). we define $F(t,x) \coloneqq E_{t_0}(z_0)^\intercal \nabla \log p_t(x) \in \mathbb{R}^{d-d^*}$ with E_{t_0} being the orthonormal basis of $N_{z_0}(\mathcal{R}_{t_0})$. Since $z_0 \in \mathcal{R}_{t_0}$, we have $F(t_0, z_0) = 0$. Taking partial derivatives of F and evaluating at (t_0, z_0), we get

$$\begin{aligned}
\partial_t F(t_0, z_0) &= E_{t_0}(z_0)^\intercal \partial_t \nabla \log p_t(z_0)|_{t=t_0}, \\
\nabla F(t_0, z_0) &= E_{t_0}(z_0)^\intercal \nabla^2 \log p_{t_0}(z_0).
\end{aligned}$$

Pick a normal gauge curve $t \mapsto z(t) \in \mathcal{R}_t$ with $z(t_0) = z_0$ and $\partial_t z(t)|_{t=t_0} \in N_{z_0}(\mathcal{R}_{t_0})$. We have $F(z, z(t)) = 0$. Taking derivative wrt. t to both sides of the equation, we get

$$\begin{aligned}
0 &= \partial_t F(t_0, z_0) + \nabla F(t_0, z_0) \partial_t z \\
&= E_{t_0}(z_0)^\intercal \partial_t \nabla \log p_t(z_0)|_{t=t_0} + E_{t_0}(z_0)^\intercal \nabla^2 \log p_{t_0}(z_0) \partial_t z(t)|_{t=t_0} \\
&= E_{t_0}(z_0)^\intercal \partial_t \nabla \log p_t(z_0)|_{t=t_0} + E_{t_0}(z_0)^\intercal \nabla^2 \log p_{t_0}(z_0) E_{t_0}(z_0) \theta_0,
\end{aligned}$$

In the last identity, due to the fact that $\partial_t z(t)|_{t=t_0} \in N_{z_0}(\mathcal{R}_{t_0})$, we write $\partial_t z(t)|_{t=t_0} = E_{t_0}(z_0)\theta_0$ for some $\theta_0 \in \mathbb{R}^{d-d^*}$. According to the definition of $\mathcal{R}_{t_0}$, all eigenvalues of $E_{t_0}(z_0)^\intercal \nabla^2 \log p_{t_0}(z_0) E_{t_0}(z_0) \in \mathbb{R}^{(d-d^*)\times(d-d^*)}$ are less than $-\beta_{t_0}$. Therefore, $E_{t_0}(z_0)^\intercal \nabla^2 \log p_{t_0}(z_0) E_{t_0}(z_0)$ is invertible and

$$\|(E_{t_0}(z_0)^\intercal \nabla^2 \log p_{t_0}(z_0) E_{t_0}(z_0))^{-1}\| \leq 1/\beta_{t_0}.$$

Hence the normal velocity field $v_{t_0}(z_0)$ is unique and

$$\begin{aligned} v_{t_0}(z_0) &= E_{t_0}(z_0)\theta_0 \\ &= E_{t_0}(z_0)(E_{t_0}(z_0)^\intercal \nabla^2 \log p_{t_0}(z_0) E_{t_0}(z_0))^{-1} E_{t_0}(z_0)^\intercal \partial_t \nabla \log p_t(z_0)|_{t=t_0} \\ &= (E_{t_0}(z_0)^\intercal \nabla^2 \log p_{t_0}(z_0) E_{t_0}(z_0))^{-1} E_{t_0}(z_0)^\intercal \partial_t \nabla \log p_t(z_0)|_{t=t_0}. \end{aligned}$$

Therefore, the Lemma is proved by varying t_0 and z_0. □

Lemma D.10 (Expression of ridge motion)**.** *Let $E_t \in \mathbb{R}^{d\times(d-d^*)}$ be the normal eigen-matrix in the definition of $\mathcal{R}_t$, i.e., $\mathcal{R}_t = \{x \in \mathbb{R}^d \mid E_t(x)\nabla \log p_t(x) = 0, , \lambda_{d^*+1} \le -\beta_t\}$. Then we have*

$$E_t(z)^\intercal \partial_t \nabla \log p_t(z) = \frac{1}{h_t} E_t(z)^\intercal \partial_t m(t,z).$$

Proof of Lemma D.10. According to Tweedie's formula, we have

$$\begin{aligned} & E_t(x)\nabla \log p_t(x) = \frac{1}{h_t} E_t(x)(m(t,x)-x), \\ \implies & E_t(x)\partial_t \nabla \log p_t(x) = \frac{1}{h_t} E_t(x)\partial_t m(t,x) - \frac{\partial_t h_t}{h_t^2} E_t(x)(m(t,x)-x). \end{aligned} \tag{18}$$

For $x = z \in \mathcal{R}_t$, we have

$$E_t(x)\nabla \log p_t(x) = \frac{1}{h_t} E_t(x)(m(t,x)-x) = 0.$$

Therefore, the last term in (18) cancels. The Lemma is proved. □

Lemma D.11. *For any $z \in \mathcal{R}_t$, let $E_t(z)$ be the eigen-matrix in Definition 3.2. We have*

$$\|E_t(z)^\intercal \partial_t m(t,z)\| \lesssim (1+\dot{h}_t) a_t R.$$

Proof of Lemma D.11. By Lemma D.7,

$$\begin{aligned} & \partial_t m(t,z) \\ &= \partial_t (h_t \nabla \log p_t(z) + h_t z) \\ &= \dot{h}_t \nabla \log p_t(z) + h_t \nabla (\frac{\partial_t p_t(z)}{p_t(z)}) + \dot{h}_t z \\ &= \dot{h}_t \nabla \log p_t(z) + h_t \nabla (\frac{z\cdot\nabla p_t(z) + \Delta p_t(z) + d p_t(z)}{p_t(z)}) + \dot{h}_t z \\ &= \dot{h}_t \nabla \log p_t(z) + h_t \nabla^2 \log p_t(z) z + d h_t \nabla \log p_t(z) + h_t \nabla (\frac{\Delta p_t(z)}{p_t(z)}) + \dot{h}_t z \\ &= (\dot{h}_t + d h_t)\nabla \log p_t(z) + h_t \nabla^2 \log p_t(z) z + \dot{h}_t z \\ &\quad + h_t(\mathrm{cont}_{23} \nabla^3 \log p_t(z) + 2\nabla^2 \log p_t(z) \nabla \log p_t(z)). \end{aligned}$$

Since $E_t(z)^\intercal \nabla \log p_t(x) = 0$ and $E_t(z)^\intercal \nabla^2 \log p_t(z) \nabla \log p_t(z) = 0$ for all $z \in \mathcal{R}_t$, we have

$$E_t(z)^\intercal \partial_t m(t,z) = h_t E_t(z)^\intercal \nabla^2 \log p_t(z) z + \dot{h}_t E_t(z)^\intercal z + h_t E_t(z)^\intercal \mathrm{cont}_{23} \nabla^3 \log p_t(z).$$

Next we bound the three terms on the RHS. First, according to Lemma D.7 and the fact that $E_t(z)^\intercal \nabla \log p_t(z) = 0$,

$$\|E_t(z)^\intercal z\| = \|E_t(z)^\intercal m(t,z)\| \le \|m(t,z)\| \le a_t R.$$

Next, according to Lemma D.7, $\nabla^2 \log p_t = -\frac{1}{h_t} I_d + \frac{\Sigma(t,z)}{h_t^2}$ with $\Sigma(t,z) \succeq 0$. Therefore, all negative eigenvalues of $\nabla^2 \log p_t$ are at least $-\frac{1}{h_t}$. On the other hand, $E_t(z)^\intercal$ only preserve eigenvalues that are smaller than $-\beta_t$. Therefore, $\nabla^2 \log p_t(z)$ only contributes eigenvalues that are smaller than $-\beta_t$, hence negative, in $E_t(z)^\intercal \nabla^2 \log p_t(z) z$. We have

$$\|E_t(z)^\intercal \nabla^2 \log p_t(z) z\| \le \frac{1}{h_t} \|E_t(z)^\intercal z\| \le \frac{a_t R}{h_t}.$$

Regarding the last term, according to Lemma D.5,

$$\|E_t(z)^\intercal \mathrm{cont}_{23}\nabla^3 \log p_t(z)\| = \|E_t(z)^\intercal(\nabla^3 \log p_t(z) : I)\| \le \sqrt{d}\|\nabla^3 \log p_t(z)\| = \mathcal{O}(\frac{\sqrt{d}\theta_t R^3}{h_t^3}).$$

Since θ_t can be arbitrarily chosen with order $\exp(-o(h_t^{-1}))$, the last term is negligible in the final order estimation. Hence we proved Lemma D.11. □

E. Analysis of Stage 1

In this section, we analyze the first stage: the reverse-time inference process enters the tube neighborhood of the log-density ridge with high probability. The formal version of Theorem 3.5 is derived from considering exactly the reverse OU process as the reference process:

$$\mathrm{d}X_t^{\leftarrow} = \big((1-\frac{2}{h_{T-t}})X_t^{\leftarrow} + \frac{2}{h_{T-t}} m(T-t, X_t^{\leftarrow})\big)\mathrm{d}t + \sqrt{2}\mathrm{d}B_t^{\leftarrow}, \quad X_0^{\leftarrow} \sim p_T,$$

and $X_t^{\leftarrow} \sim p_{T-t}$ for all $0 \le t \le T_\delta$. We define the entering time of $X_t^{\leftarrow}$:

$$t_{\mathrm{in}}^{\leftarrow} := \inf\{0 \le t \le T-\delta | X_t^{\leftarrow} \in \mathcal{T}_{T-t}(\rho_{T-t})\}.$$

And we also analyze the reverse-time inference dynamics Y_t with

$$t_{\mathrm{in}} := \inf\{0 \le t \le T-\delta \mid Y_t \in \mathcal{T}_{T-t}(\rho_{T-t})\}.$$

Then the formal Theorem that describing our stage 1 states as follows:

Theorem E.1. *Under Assumption 3.1, we have*

(1) $\mathbb{P}(t_{\mathrm{in}}^{\leftarrow} \le T-\delta) \ge 1-e_\delta$;

(2) $\mathbb{P}(t_{\mathrm{in}} \le T-\delta) \ge 1-e_\delta-\varepsilon(T)$;

(3) $\mathbb{P}(\tilde{t}_{\mathrm{in}} \le T-\delta) \ge 1-e_\delta-\varepsilon(T)-\sqrt{\varepsilon_A(T,\delta)/8}$,

where $e_\delta = h_\delta^\zeta$ *for any* $\zeta > 0$ *and* $e_\delta \to 0$ *at any polynomial order as* $\delta \to 0^+$. $\varepsilon(T) = \frac{a_T}{2}(\frac{\sqrt{d}}{\sqrt{h_T}} + R) \to 0$ *exponentially fast as* $T \to \infty$.

Proof of Theorem E.1. First, for the exact reverse OU process, we have

$$\mathbb{P}(X_{T-\delta}^{\leftarrow} \in \mathcal{T}_\delta(\rho_\delta)) = \mathbb{P}(X_\delta \in \mathcal{T}_\delta(\rho_\delta)) = \mathbb{P}(\mathrm{dist}(X_\delta, \mathcal{R}_\delta) \le \rho_\delta) \ge \mathbb{P}(\mathrm{dist}(a_\delta X_0, \mathcal{R}_\delta) + \sqrt{h_\delta}\|z\| \le \rho_\delta).$$

We will first show $\mathrm{dist}(a_\delta X_0, \mathcal{R}_\delta)$ is very small when $\delta \ll 1$, and then apply the concentration of d-dimensional Gaussian to bound the probability.

According to Lemma E.2, if δ is small enough such that $h_\delta^{-1} > \frac{2}{a_\delta^2\Delta^2}\ln(\frac{4a_\delta^2R^2(n-1)}{\rho_\delta})$, then

$$\mathrm{dist}(a_\delta X_0, \mathcal{R}_\delta) \le \|z_i - a_\delta x_0^{(i)}\| \le \rho_\delta/2.$$

Then we have

$$\mathbb{P}(X_{T-\delta}^{\leftarrow} \in \mathcal{T}_\delta(\rho_\delta)) \ge \mathbb{P}(\sqrt{h_\delta}\|z\| \le \rho_\delta) = \mathbb{P}(\|z\|^2 \le \rho_\delta^2/h_\delta)$$

According to the order estimation of r_t in Proposition D.1, we can pick δ small such that $\rho_\delta^2/h_\delta \le d + 2\sqrt{d\zeta\log(1/h_\delta)} + 2\zeta\log(1/h_\delta)$ for any $\zeta > 0$. Then according to the LMI inequality (Moshksar, 2024), we have

$$\mathbb{P}(X_{T-\delta}^{\leftarrow} \in \mathcal{T}_\delta(\rho_\delta)) \ge \mathbb{P}(\|z\|^2 \le d + 2\sqrt{d\zeta\log(1/h_\delta)} + 2\zeta\log(1/h_\delta)) \ge 1 - h_\delta^\zeta.$$

Next, for the reverse-time inference process Y_t. Notice that Y_t and $X_t^{\leftarrow}$ have the same generator but with different initializations: $Y_0 \sim \mathcal{N}(0, I_d)$ while $X_0^{\leftarrow} \sim p_T$. Therefore, we have

$$\begin{aligned}|\mathbb{P}(Y_{T-\delta} \in \mathcal{T}_\delta(\rho_\delta)) - \mathbb{P}(X_{T-\delta}^{\leftarrow} \in \mathcal{T}_\delta(\rho_\delta))| &\le \mathrm{TV}(\mathrm{Law}(Y_{T-\delta}), p_\delta) \le \mathrm{TV}(\mathcal{N}(0, I_d), p_T) \\ &\le \sqrt{\frac{\mathrm{KL}(p_T|\mathcal{N}(0, I_d))}{2}} \le \frac{a_T}{2}(\frac{\sqrt{d}}{\sqrt{h_T}} + R),\end{aligned}$$

where the second inequality follows from data processing inequality. The third inequality follows from Pinsker's inequality. The last inequality follows from property of OU process, see He et al. (2024, Proposition C.1). Therefore, we proved

$$\mathbb{P}(Y_{T-\delta} \in \mathcal{T}_\delta(\rho_\delta)) \ge \mathbb{P}(X_{T-\delta}^{\leftarrow} \in \mathcal{T}_\delta(\rho_\delta)) - \frac{a_T}{2}(\frac{\sqrt{d}}{\sqrt{h_T}} + R) \ge 1 - h_\delta^\zeta - \frac{a_T}{2}(\frac{\sqrt{d}}{\sqrt{h_T}} + R).$$

Last, for the inference process $\tilde{Y}_t$, except for initialization error, there are extra trajectory error from $X_t^{\leftarrow}$ due to the approximate posterior mean m_A. We have

$$|\mathbb{P}(\tilde{Y}_{T-\delta} \in \mathcal{T}_\delta(\rho_\delta)) - \mathbb{P}(X_{T-\delta}^{\leftarrow} \in \mathcal{T}_\delta(\rho_\delta))| \le \mathrm{TV}(\mathrm{Law}(\tilde{Y}_{T-\delta}), p_\delta) \le \sqrt{\frac{\mathrm{KL}(\mathbb{P}^{\leftarrow}|\tilde{\mathbb{P}})}{2}},$$

where $\mathbb{P}^{\leftarrow}$ is the path measure of $X_t^{\leftarrow}$ and $\tilde{\mathbb{P}}$ is the path measure of $\tilde{Y}_t$. Then we can apply the traditional analysis of the inference process via Girsanov's Theorem. We get

$$\begin{aligned}&|\mathbb{P}(\tilde{Y}_{T-\delta} \in \mathcal{T}_\delta(\rho_\delta)) - \mathbb{P}(X_{T-\delta}^{\leftarrow} \in \mathcal{T}_\delta(\rho_\delta))| \\ &\le \sqrt{\frac{\mathrm{KL}(p_T|\mathcal{N}(0, I_d))}{2} + \frac{1}{8}\int_\delta^T h_t^{-2}\mathbb{E}[\|m(t, X_t) - m_A(t, X_t)\|^2]\mathrm{d}t} \\ &\le \frac{a_T}{2}(\frac{\sqrt{d}}{\sqrt{h_T}} + R) + \sqrt{\frac{1}{8}\int_\delta^T h_t^{-2}\mathbb{E}[\|m(t, X_t) - m_A(t, X_t)\|^2]\mathrm{d}t}.\end{aligned}$$

The statements follows from transferring the probability to the defined stopping times. □

Lemma E.2. *Under Assumption 3.1, when $\delta \ll 1$, for each $i \in [n]$, there exists a point $z_i \in \mathcal{R}_\delta$ such that*

$$\|z_i - a_\delta x_0^{(i)}\| \le 2a_\delta R(n-1)\exp(-\frac{a_\delta^2\Delta^2}{2h_\delta}).$$

Proof of Lemma E.2. According to Lemma D.7, we know

$$\begin{aligned}\nabla \log p_\delta(x) &= -\frac{x}{h_\delta} + \frac{1}{h_\delta}\sum_{i=1}^n \mathrm{Softmax}(-\frac{\|x - a_\delta x_0\|^2}{2h_\delta})_i a_\delta x_0^{(i)} \\ \nabla^2 \log p_\delta(x) &= -\frac{1}{h_\delta}I_d + \frac{1}{h_\delta^2}\sum_{i=1}^n \mathrm{Softmax}(-\frac{\|x - a_\delta x_0\|^2}{2h_\delta})_i (a_\delta x_0^{(i)} - m(\delta, x))(a_\delta x_0^{(i)} - m(\delta, x))^\intercal.\end{aligned}$$

Next, we find critical points of $\nabla \log p_\delta(x)$ and show that they are on the log-density ridge $\mathcal{R}_\delta$.

According to the expression of $\nabla \log p_\delta(x)$, the critical points are fixed points of

$$x = \sum_{i=1}^n \mathrm{Softmax}(-\frac{\|x - a_\delta x_0\|^2}{2h_\delta})_i a_\delta x_0^{(i)} = m(\delta, x).$$

We claim: for each $i \in [n]$, within $B(a_\delta x_0^{(i)}, a_\delta\Delta/4)$, there exists a unique fixed point, denoted as z_i. We prove the claim by the contraction mapping theorem. First, $B(a_\delta x_0^{(i)}, a_\delta\Delta/4) \subset \mathbb{R}^d$ is a closed ball. Second, for all $x \in B(a_\delta x_0^{(i)}, a_\delta\Delta/4)$,

for all $j \neq i$, we have

$$\frac{\text{Softmax}(-\frac{\|x-a_\delta x_0\|}{2h_\delta})_j}{\text{Softmax}(-\frac{\|x-a_\delta x_0\|}{2h_\delta})_i} = \exp(-\frac{\|x-a_\delta x_0^{(j)}\|}{2h_\delta} + \frac{\|x-a_\delta x_0^{(i)}\|}{2h_\delta}) \leq \exp(-\frac{a_\delta^2\Delta^2}{2h_\delta}),$$
$$\implies \sum_{j\neq i} \text{Softmax}(-\frac{\|x-a_\delta x_0\|}{2h_\delta})_j \leq (n-1)\exp(-\frac{a_\delta^2\Delta^2}{2h_\delta}).$$

Hence, we can show $m(\delta,\cdot): B(a_\delta x_0^{(i)}, a_\delta\Delta/4) \to B(a_\delta x_0^{(i)}, a_\delta\Delta/4)$, i.e., for all $x \in B(a_\delta x_0^{(i)}, a_\delta\Delta/4)$,

$$\begin{aligned}\|m(\delta,x) - a_\delta x_0^{(i)}\| &\leq \sum_{j\neq i} \text{Softmax}(-\frac{\|x-a_\delta x_0\|}{2h_\delta})_j a_\delta \|x_0^{(j)} - x_0^{(i)}\| \\ &\leq 2a_\delta R(n-1)\exp(-\frac{a_\delta^2\Delta^2}{2h_\delta}) < a_\delta\Delta/4,\end{aligned}$$

given the early stopping time is small enough such that $h_\delta^{-1} > \frac{2}{a_\delta^2\Delta^2}\ln(\frac{8R(n-1)}{\Delta})$. Meanwhile, for $x \in B(a_\delta x_0^{(i)}, a_\delta\Delta/4)$, similar to the proof of Lemma D.5,

$$\begin{aligned}\|\nabla m(\delta,x)\| &\leq \frac{1}{h_\delta}\sum_{j\neq i} \text{Softmax}(-\frac{\|x-a_\delta x_0\|}{2h_\delta})_j a_\delta^2 \|x_0^{(j)} - x_0^{(i)}\|^2 \\ &\leq \frac{a_\delta^2 R^2(n-1)}{h_\delta}\exp(-\frac{a_\delta^2\Delta^2}{2h_\delta}) < 1,\end{aligned}$$

given the early stopping time is small enough such that $h_\delta^{-1} > \frac{2}{a_\delta^2\Delta^2}\ln(\frac{a_\delta^2R^2(n-1)}{h_\delta})$. Therefore, according to the contraction mapping theorem, there exists a unique fixed point $z_i \in B(a_\delta x_0^{(i)}, a_\delta\Delta/4)$. At each z_i, according to Lemma D.7, we have

$$\begin{aligned}\nabla^2 \log p_\delta(z_i) &= -\frac{1}{h_\delta} I_d + \frac{\text{Cov}(U(\delta,x))}{h_\delta^2} \\ &\preceq -\frac{1}{h_\delta} I_d + \frac{1}{h_\delta^2} a_\delta^2 R^2(n-1)\exp(-\frac{a_\delta^2\Delta^2}{2h_\delta}) I_d \\ &\preceq -\frac{1}{2h_\delta} I_d,\end{aligned}$$

given the early stopping time is small enough such that $h_\delta^{-1} > \frac{2}{a_\delta^2\Delta^2}\ln(\frac{a_\delta^2R^2(n-1)}{2h_\delta})$. Therefore, we proved that $z_i \in \mathcal{R}_\delta$ with $\beta_\delta = \Theta(\frac{1}{h_\delta})$. Last, we estimate the distance from z_i to $a_\delta x_0^{(i)}$.

$$\begin{aligned}\|z_i - a_\delta x_0^{(i)}\| = \|m(t,z_i) - x_0^{(i)}\| &\leq \sum_{j\neq i} \text{Softmax}(-\frac{\|z_i-a_\delta x_0\|}{2h_\delta})_j a_\delta \|x_0^{(j)} - x_0^{(i)}\| \\ &\leq 2a_\delta R(n-1)\exp(-\frac{a_\delta^2\Delta^2}{2h_\delta}).\end{aligned}$$

□

F. Analysis of Stage 2

In this section, we analyze the stage 2 - align along normal directions. We start from stating the formal version of Theorem 3.6 which includes the dynamical property of the squared normal distance to the ridge.

Theorem F.1. *Under Assumption 3.1, define* $\kappa_{s,t} = 2\int_s^t(\beta_{T-u}-1)\mathrm{d}u$, $B_{s,t}(d,R) = \int_s^t e^{-\kappa_{u,t}}(\rho_{T-u}R + d)\mathrm{d}u$ *and* $e_A^\perp(t,x) := P^\perp(\Pi_t(x))e_A(t,x)$ *with* $e_A = m_A - m$. *Then for any* $t \in [\tilde{t}_{\text{in}}, T-\delta]$,

$$\mathbb{E}[D_{T-t}(\tilde{Y}_t)] \lesssim e^{-\kappa_{\tilde{t}_{\text{in}},t}}\rho_{T-\tilde{t}_{\text{in}}} + B_{\tilde{t}_{\text{in}},t}(d,R) + \int_{\tilde{t}_{\text{in}}}^t e^{-\kappa_{u,t}}\frac{\mathbb{E}[\|e_A^\perp(T-u,\tilde{Y}_u)\|^2]}{h_{T-u}^2\beta_{T-u}}\mathrm{d}u.$$

Furthermore, picking picking $\beta_t = c/h_t$ for any $c \in [\frac{1}{2}, 1)$[3], when $\delta \ll 1$, we have

$$\mathbb{E}[D_\delta(\tilde{Y}_{T-\delta})] = \mathcal{O}(d\delta^c + \delta^c \int_{\tilde{t}_{\text{in}}}^{T-\delta} h_{T-u}^{-1-c}\, \mathbb{E}[\|e_A^\perp(T-u, \tilde{Y}_u)\|^2]\mathrm{d}u).$$

We will prove Theorem F.1 and Theorem 3.6 naturally follows from it. The proof relies on a property of the square-normal distance, which simply follows from the log-density ridge property proved in Appendix D.

Corollary F.2. *As a consequence of Proposition C.1, the following properties hold for the square-distance function $D_t(x) \coloneqq \|x - \Pi_t(x)\|^2$:*

(1) $\mathcal{D}_t \in C^2(\mathcal{T}_t(\rho_t))$ and $\nabla D_t(x) = 2n_t(x)$;

(2) $\sup_x \|\nabla^2 D_t(x)\| \le \frac{2}{1-\rho_t/r_t}$. As a consequence, $\sup_x \Delta D_t(x) \le \frac{2d}{1-\rho_t/r_t}$.

We now prove Theorem F.1 for both Y_t and $\tilde{Y}_t$, given below

$$\mathrm{d}Y_t = \underbrace{(Y_t - \frac{2}{h_{T-t}}Y_t + \frac{2}{h_{T-t}}m(T-t, Y_t))\,\mathrm{d}t}_{\coloneqq b(T-t, Y_t)} + \sqrt{2}\mathrm{d}\bar{B}_t, \quad Y_0 \sim \mathcal{N}(0, I_d),$$
$$\mathrm{d}\tilde{Y}_t = \underbrace{(\tilde{Y}_t - \frac{2}{h_{T-t}}\tilde{Y}_t + \frac{2}{h_{T-t}}m_A(T-t, \tilde{Y}_t))\,\mathrm{d}t}_{\coloneqq b_A(T-t, \tilde{Y}_t)} + \sqrt{2}\mathrm{d}\tilde{B}_t, \quad \tilde{Y}_0 \sim \mathcal{N}(0, I_d).$$

Note that for Y_t, it is differed from exact reverse-time OU only in initialization. Therefore, we would like to highlight the relation between reverse OU-dynamics and the log-density ridge geometry through analysis of Y_t. The gap between Y_t and $\tilde{Y}_t$ lies in the approximation error of the posterior mean. Therefore, our analysis for $\tilde{Y}_t$ will highlight the effect of posterior mean approximation error.

Proof of Theorem F.1. For any $t \in [0, T-\delta]$, $x \in \mathcal{T}_{T-t}(\rho_t)$, recall that $D_{T-t}(x) = \|n_{T-t}(x)\|^2$ and $n_{T-t}(x) = x - \Pi_{T-t}(x)$. Once the processes enter $\mathcal{T}_{T-t}(\rho_t)$, we can track the evolution of D_{T-t} via Itô's formula. For the reverse process Y_t, we have

$$\begin{aligned}\mathrm{d}D_{T-t}(Y_t) = \big(&\partial_t D_{T-t}(Y_t) + 2\langle n_{T-t}(Y_t), b(T-t, Y_t)\rangle + 2\mathrm{trace}(\nabla^2 D_{T-t}(Y_t))\big)\mathrm{d}t \\ &+ 2\sqrt{2}\langle n_{T-t}(Y_t), \mathrm{d}\bar{B}_t\rangle,\end{aligned}$$

where we have the following estimations for the drift terms:

(1) According to Corollary F.2, $2\mathrm{trace}(\nabla^2 D_{T-t}(Y_t)) \le \frac{4d}{1-\rho_{T-t}/r_{T-t}}$. Picking $\rho_t = r_t/2$ for all t and we get $2\mathrm{trace}(\nabla^2 D_{T-t}(Y_t)) \le 8d$.

(2) According to Remark D.2,

$$\begin{aligned}\partial_t D_{T-t}(Y_t) &= -\partial_s D_s(Y_t)|_{s=T-t} = 2\langle n_{T-t}(Y_t), \partial_s \Pi_s(Y_t)|_{s=T-t}\rangle \\ &\le 2\rho_{T-t} \sup_{x \in \mathcal{T}_{T-t}(\rho_{T-t})} \|\partial_s \Pi_s(x)|_{s=T-t}\| \coloneqq 2\rho_{T-t} S_{T-t},\end{aligned}$$

and $S_{T-t} = \mathcal{O}(R)$ when $T - t \to 0^+$.

(3) if we denote $z = z_{T-t} \coloneqq \Pi_{T-t}(Y_t)$, $n_{T-t}(Y_t) \perp T_z(\mathcal{R}_{T-t})$ and we have

$$\begin{aligned}2\langle n_{T-t}(Y_t), b(T-t, Y_t)\rangle &= 2\langle n_{T-t}(Y_t), Y_t\rangle + 4\langle n_{T-t}(Y_t), \nabla \log p_{T-t}(Y_t)\rangle \\ &= 2\|n_{T-t}(Y_t)\|^2 + 4\langle n_{T-t}(Y_t), \nabla \log p_{T-t}(Y_t)\rangle,\end{aligned}$$

[3] c can be chosen arbitrarily between $[\frac{1}{2}, 1)$ due to the property of $\nabla^2 \log p_t(x)$ as explained in Remark D.6.

In the last term, we can write $Y_t = z + n_{T-t}(Y_t)$ and expand $\nabla \log p_{T-t}(Y_t)$ at $z \in \mathcal{R}_{T-t}$. Then we have

$$\begin{aligned}
&\langle n_{T-t}(Y_t), \nabla \log p_{T-t}(Y_t)\rangle \\
=&\langle n_{T-t}(Y_t), \nabla \log p_{T-t}(z)\rangle + \langle n_{T-t}(Y_t), \nabla^2 \log p_{T-t}(z) n_{T-t}(Y_t)\rangle \\
&+ \frac{1}{2}\langle n_{T-t}(Y_t), \nabla^3 \log p_{T-t}(z') n_{T-t}(Y_t)^{\otimes 2}\rangle,
\end{aligned}$$

where

$$\begin{aligned}
&\langle n_{T-t}(Y_t), \nabla \log p_{T-t}(z)\rangle = 0, && \text{definition of } \mathcal{R}_{T-t},\\
&\langle n_{T-t}(Y_t), \nabla^2 \log p_{T-t}(z) n_{T-t}(Y_t)\rangle \\
=&\langle n_{T-t}(Y_t), P^{\perp}(z)\nabla^2 \log p_{T-t}(z) n_{T-t}(Y_t)\rangle \le -\beta_{T-t}\|n_{T-t}(Y_t)\|^2, && \text{definition of } \mathcal{R}_{T-t},\\
&\langle n_{T-t}(Y_t), \nabla^3 \log p_{T-t}(z') n_{T-t}(Y_t)^{\otimes 2}\rangle \\
\le& \sup_x \|\nabla \log p_{T-t}(x)\| \|n_{T-t}(Y_t)\|^3 \le \frac{80 a_{T-t}^3 R^3}{h_{T-t}^3}\rho_{T-t}\|n_{T-t}(Y_t)\|^2 && \text{Lemma } D.5\\
\le& \frac{a_{T-t}^3 \beta_{T-t}}{2} D_{T-t}(Y_t) && \text{Proposition } D.1
\end{aligned}$$

The last identity follows from Proposition D.1 by picking θ_t such that $r_t = \beta_t h_t^3 R^{-3}/80$ and $\rho_t = r_t/2$. Combining the above inequalities, we have

$$\langle n_{T-t}(Y_t), \nabla \log p_{T-t}(Y_t)\rangle \le -\beta_{T-t}(1 - a_{T-t}^3/4)\|n_{T-t}(Y_t)\|^2 \le -\frac{3}{4}\beta_{T-t} D_{T-t}(Y_t).$$

Therefore, we have

$$2\langle n_{T-t}(Y_t), b(T-t, Y_t)\rangle \le -(3\beta_{T-t} - 2) D_{T-t}(Y_t).$$

Combining all the estimations and taking expectations of D_{T-t}, we obtain the following inequality

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}[D_{T-t}(Y_t)] \le -(3\beta_{T-t} - 2)\mathbb{E}[D_{T-t}(Y_t)] + 2\rho_{T-t} S_{T-t} + 8d.$$

Last, apply Gronwall's inequality, for any $t_{\text{in}} \in (0, T-\delta)$ and $t \in (t_{\text{in}}, T-\delta]$, we obtain

$$\begin{aligned}
\mathbb{E}[D_{T-t}(Y_t)] \le& \exp\big(-\int_{t_{\text{in}}}^t 3\beta_{T-u} - 2\mathrm{d}u\big)\mathbb{E}[D_{t-t_{\text{in}}}(Y_{t_{\text{in}}})] \\
&+ \int_{t_{\text{in}}}^t \exp\big(-\int_u^t 3\beta_{T-s} - 2\mathrm{d}s\big)\big(2\rho_{T-u} S_{T-u} + 8d\big)\mathrm{d}u.
\end{aligned}$$

When $t = T - \delta$, $\beta_t = c/h_t$ and $\delta \ll 1$, we have

$$\int_{t_{\text{in}}}^{T-\delta} \exp\big(-\int_u^{T-\delta} 3\beta_{T-s} - 2\mathrm{d}s\big)\big(2\rho_{T-u} S_{T-u} + 8d\big)\mathrm{d}u = \mathcal{O}(d\delta^{\frac{3c}{2}}).$$

For the reverse-time inference process $\tilde{Y}_t$, we have

$$\begin{aligned}
\mathrm{d}D_{T-t}(\tilde{Y}_t) =& \big(\partial_t D_{T-t}(\tilde{Y}_t) + 2\langle n_{T-t}(\tilde{Y}_t), b_A(T-t, \tilde{Y}_t)\rangle \\
&+ 2\mathrm{trace}(\nabla^2 D_{T-t}(\tilde{Y}_t))\big)\mathrm{d}t + 2\sqrt{2}\langle n_{T-t}(\tilde{Y}_t), \mathrm{d}\tilde{B}_t\rangle.
\end{aligned}$$

Notice that the only difference to that of Y_t is the error $e_A(t,\cdot) = m(t,\cdot) - m_A(t,\cdot)$ within the normal space, i.e., $e_A^{\perp}(t,x) \coloneqq P^{\perp}(\Pi_t(x))e_A(t,x)$. Similarly, we have the inequality

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}[D_{T-t}(\tilde{Y}_t)] \\
\le& -(3\beta_{T-t} - 2)\mathbb{E}[D_{T-t}(\tilde{Y}_t)] + 2\rho_{T-t} S_{T-t} + 8d + 2\mathbb{E}[\langle n_{T-t}(\tilde{Y}_t), \frac{1}{h_{T-t}} e_A(T-t, \tilde{Y}_t)\rangle] \\
\le& -2(\beta_{T-t} - 1)\mathbb{E}[D_{T-t}(\tilde{Y}_t)] + 2\rho_{T-t} S_{T-t} + 8d + \frac{\mathbb{E}[\|e_A^{\perp}(T-t, \tilde{Y}_t)\|^2]}{h_{T-t}^2 \beta_{T-t}},
\end{aligned}$$

where the last inequality follows from Young's inequality. Therefore, according to the Gronwall's inequality,

$$\mathbb{E}[D_{T-t}(\tilde{Y}_t)] \le \exp\big(-2\int_{\tilde{t}_{\text{in}}}^{t} \beta_{T-u} - 1\mathrm{d}u\big)\mathbb{E}[D_{t-\tilde{t}_{\text{in}}}(\tilde{Y}_{\tilde{t}_{\text{in}}})] + \int_{\tilde{t}_{\text{in}}}^{t} \exp\big(-2\int_{u}^{t} \beta_{T-s} - 1\mathrm{d}s\big)\big(2\rho_{T-u}S_{T-u} + \frac{\mathbb{E}[\|e_A^{\perp}(T-u,\tilde{Y}_u)\|^2]}{h_{T-u}^2\beta_{T-u}} + 8d\big)\mathrm{d}u.$$

When $t = T-\delta$, $\beta_t = c/h_t$ and $\delta \ll 1$, we have

$$\begin{aligned}
&\int_{\tilde{t}_{\text{in}}}^{T-\delta} \exp\big(-2\int_{u}^{T-\delta} \beta_{T-s} - 1\mathrm{d}s\big)\mathrm{d}u = \mathcal{O}(\delta^c),\\
&\int_{\tilde{t}_{\text{in}}}^{T-\delta} \exp\big(-2\int_{u}^{T-\delta} \beta_{T-s} - 1\mathrm{d}s\big)h_{T-u}^{-2}\beta_{T-u}^{-1}\mathbb{E}[\|e_A^{\perp}(T-u,\tilde{Y}_u)\|^2]\mathrm{d}u\\
=\ &\mathcal{O}(\delta^c\int_{\tilde{t}_{\text{in}}}^{T-\delta} h_{T-u}^{-1-c}\mathbb{E}[\|e_A^{\perp}(T-u,\tilde{Y}_u)\|^2]\mathrm{d}u).
\end{aligned}$$

□

G. Analysis of Stage 3

In this section, we analyze Stage 3, namely the tangential sliding behavior of both the ideal reverse process Y_t and the learned reverse process $\tilde{Y}_t$ relative to the log-density ridge geometry. We first state a formal dynamical version of Theorem 3.7, which controls the evolution of the squared tangential residual along inference. The terminal-time estimate stated in Theorem 3.7 in the main text follows directly by integrating this formal bound.

Theorem G.1. *Under Assumption 3.1, for any $t \in [\tilde{t}_{\text{in}}, T-\delta]$, define $e_A^{\parallel,i}(t,x) = (U_t^{(i)})^{\intercal} e_A(t,x)$ with $e_A = m_A - m$. If $\tilde{Y}_t \in \mathcal{B}_{T-t}^{(i)}(\theta_{T-t})$, then we have*

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}[\|\tilde{u}_t^{(i)}\|^2] \le -(\frac{1}{h_{T-t}} - 2)\mathbb{E}[\|\tilde{u}_t^{(i)}\|^2] + \tilde{\epsilon}_{T-t}^{(i)} + \frac{4}{h_{T-t}}\|e_A^{\parallel,i}(T-t,\tilde{Y}_t)\|^2,$$

where $\tilde{\epsilon}_{T-t}^{(i)} = \mathcal{O}(d)$. Furthermore, with $\delta \ll 1$, we have

$$\mathbb{E}[\|\tilde{u}_{T-\delta}^{(i)}\|^2] = \mathcal{O}(d\sqrt{\delta} + \sqrt{\delta}\int_{\tilde{t}_{\text{in}}}^{T-\delta} h_{T-u}^{-\frac{3}{2}}\mathbb{E}[\|e_A^{\parallel,i}(T-u,\tilde{Y}_u)\|^2]\mathrm{d}u).$$

We now introduce the objects used in the proof. Recall that the ideal and learned reverse processes satisfy

$$\begin{aligned}
\mathrm{d}Y_t &= \underbrace{(Y_t - \frac{2}{h_{T-t}}Y_t + \frac{2}{h_{T-t}}m(T-t,Y_t))\,\mathrm{d}t}_{:=b(T-t,Y_t)} + \sqrt{2}\mathrm{d}\bar{B}_t, \quad Y_0 \sim \mathcal{N}(0,I_d),\\
\mathrm{d}\tilde{Y}_t &= \underbrace{(\tilde{Y}_t - \frac{2}{h_{T-t}}\tilde{Y}_t + \frac{2}{h_{T-t}}m_A(T-t,\tilde{Y}_t))\,\mathrm{d}t}_{:=b_A(T-t,\tilde{Y}_t)} + \sqrt{2}\mathrm{d}\tilde{B}_t, \quad \tilde{Y}_0 \sim \mathcal{N}(0,I_d).
\end{aligned}$$

In Section 3.4, we introduced the tangent frame $U_{T-t}^{(i)} \in \mathbb{R}^{d\times d^*}$ at the center $m_{T-t}^{(i)}$, whose columns are the top-d^* eigenvectors of $\nabla \log p_{T-t}(m_{T-t}^{(i)})$. For trajectories lying in the tube neighborhood and the i^{th} center-dominant region, the natural residuals relative to the local center are $r_t^{(i)} = Y_t - m_{T-t}^{(i)}$ and $\tilde{r}_t^{(i)} = \tilde{Y}_t - m_{T-t}^{(i)}$. We then define the corresponding tangential coordinates $\tilde{u}_t^{(i)} := (U_{T-t}^{(i)})^{\intercal} r_t^{(i)} \in \mathbb{R}^{d^*}$ and $u_t^{(i)} := (U_{T-t}^{(i)})^{\intercal}\tilde{r}_t^{(i)} \in \mathbb{R}^{d^*}$. These are the appropriate quantities to study because Stage 3 concerns motion along the ridge rather than distance to the ridge. Strictly speaking, the ideal tangent frame wold be taken at the projected ridge point $\Pi_t(m_{T-t}^{(i)})$, not at the center $m_{T-t}^{(i)}$ itself. However, Lemma E.2 shows that

$m_{T-t}^{(i)}$ is exponentially close to $\Pi_t(m_{T-t}^{(i)})$ as $T-t \to \delta^+ \ll 1$. Therefore, $u_t^{(i)}$ and $\tilde{u}_t^{(i)}$ provide accurate local proxies for the tangential components of the residuals, while keeping the analysis explicit and tractable.

We next prove the formal tangential contraction theorem.

Proof of Theorem 3.7. Apply Itô's formula to $u_t^{(i)}$ and we get

$$\begin{aligned}
\mathrm{d}u_t^{(i)} &= \partial_t(U_{T-t}^{(i)})^\intercal(Y_t - m_{T-t}^{(i)})\mathrm{d}t + (U_{T-t}^{(i)})^\intercal\big(b(T-t,Y_t)\mathrm{d}t - \partial_t a_{T-t}x_0^{(i)}\mathrm{d}t + \sqrt{2}\mathrm{d}\bar{B}_t\big)\\
&= -(\frac{2}{h_{T-t}} - 1)u_t^{(i)}\mathrm{d}t + \partial_t(U_{T-t}^{(i)})^\intercal(Y_t - m_{T-t}^{(i)})\mathrm{d}t + \sqrt{2}(U_{T-t}^{(i)})^\intercal\mathrm{d}\bar{B}_t\\
&\quad + (U_{T-t}^{(i)})^\intercal\big(b(T-t,Y_t) - (Y_t + \frac{2}{h_{T-t}}(m_{T-t}^{(i)} - Y_t))\big)\mathrm{d}t.
\end{aligned}$$

Now apply Itô's formula again to $\|u_t^{(i)}\|^2$ and we get,

$$\begin{aligned}
\mathrm{d}\|u_t^{(i)}\|^2 &= 2\langle u_t^{(i)}, \mathrm{d}u_t^{(i)}\rangle + 2\mathrm{trace}((U_{T-t}^{(i)})^\intercal U_{T-t}^{(i)})\mathrm{d}t\\
&= -2(\frac{2}{h_{T-t}} - 1)\|u_t^{(i)}\|^2\mathrm{d}t + 2\langle u_t^{(i)}, \partial_t(U_{T-t}^{(i)})^\intercal(Y_t - m_{T-t}^{(i)})\rangle\mathrm{d}t\\
&\quad + 2\langle u_t^{(i)}, (U_{T-t}^{(i)})^\intercal\big(b(T-t,Y_t) - (Y_t + \frac{2}{h_{T-t}}(m_{T-t}^{(i)} - Y_t))\big)\rangle\mathrm{d}t\\
&\quad + 2d^*\mathrm{d}t + 2\sqrt{2}\langle u_t^{(i)}, (U_{T-t}^{(i)})^\intercal\mathrm{d}\bar{B}_t\rangle,
\end{aligned}$$

where according to Lemma G.4,

$$\begin{aligned}
&\langle u_t^{(i)}, \partial_t(U_{T-t}^{(i)})^\intercal(Y_t - m_{T-t}^{(i)})\rangle\\
&= \langle u_t^{(i)}, \partial_t(U_{T-t}^{(i)})^\intercal U_{T-t}^{(i)}u_t^{(i)}\rangle + \langle u_t^{(i)}, \partial_t(U_{T-t}^{(i)})^\intercal(I_d - U_{T-t}^{(i)}(U_{T-t}^{(i)})^\intercal)(Y_t - m_{T-t}^{(i)})\rangle\\
&\le 0 + \frac{1}{h_{T-t}}\|u_t^{(i)}\|^2 + h_{T-t}\|P(m_{T-t}^{(i)})\partial_t P(m_{T-t}^{(i)})(I_d - P(m_{T-t}^{(i)}))\|^2\|r_t^{(i)}\|^2,
\end{aligned}$$

and according to Lemma G.3 and Assumption 3.1,

$$\begin{aligned}
&\langle u_t^{(i)}, (U_{T-t}^{(i)})^\intercal\big(b(T-t,Y_t) - (Y_t + \frac{2}{h_{T-t}}(m_{T-t}^{(i)} - Y_t))\big)\rangle\\
&\le \|b(T-t,Y_t) - (Y_t + \frac{2}{h_{T-t}}(m_{T-t}^{(i)} - Y_t))\|\|Y_t - a_{T-t}x_0^{(i)}\|\\
&\le 8R^2a_{T-t}^2h_{T-t}^{-1}\theta_{T-t},
\end{aligned}$$

Therefore, taking expectations on both sides of the SDE for $\|u_t^{(i)}\|^2$, we have

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}[\|u_t^{(i)}\|^2] &\le -2(\frac{1}{h_{T-t}} - 1)\mathbb{E}[\|u_t^{(i)}\|^2] + 2h_{T-t}\|P(m_{T-t}^{(i)})\partial_t P(m_{T-t}^{(i)})(I_d - P(m_{T-t}^{(i)}))\|^2\|r_t^{(i)}\|^2\\
&\quad + 8R^2a_{T-t}^2h_{T-t}^2\eta_{T-t} + 2d^*,
\end{aligned}$$

where as $T-t \to \delta^+ \ll 1$,

(1) $-2(\frac{1}{h_{T-t}} - 1)\mathbb{E}[\|u_t^{(i)}\|^2]$ is a strict contraction term with infinite force;

(2) $2h_{T-t}\|P(m_{T-t}^{(i)})\partial_t P(m_{T-t}^{(i)})(I_d - P(m_{T-t}^{(i)}))\|^2\|r_t^{(i)}\|^2 = O(1)$:

(i) Under Assumption 3.1, $\|r_t^{(i)}\|^2 = \mathcal{O}(R^2)$;

(ii) According to the definition of $P(m_{T-t}^{(i)})$ and estimations Lemma D.5,

$$\|P(m_{T-t}^{(i)})\partial_t P(m_{T-t}^{(i)})(I_d - P(m_{T-t}^{(i)}))\| = \mathcal{O}(\theta_{T-t}\mathrm{poly}(R, h_t^{-1}))$$

for any $\theta_{T-t} = \exp(-o(\frac{1}{h_{T-t}}))$;

Combining the two estimations, we have the whole term is $\mathcal{O}(1)$.

(3) $8R^2a_{T-t}^2h_{T-t}^2\theta_{T-t} = \mathcal{O}(1)$ since we can choose $\theta_{T-t} = \exp(-o(\frac{1}{h_{T-t}}))$ as discussed in Remark D.4.

Therefore, we obtain that $\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}[\|u_t^{(i)}\|^2] \le -2(\frac{1}{h_{T-t}} - 1)\mathbb{E}[\|u_t^{(i)}\|^2] + \epsilon_{T-t}^{(i)}$ with $\epsilon_{T-t}^{(i)} = \mathcal{O}(d)$ as $t \to T^-$. By Gronwall's inequality,

$$\mathbb{E}[\|u_t^{(i)}\|^2] \le \exp\big(-2\int_{t_{\mathrm{in}}}^t \frac{1}{h_{T-u}} - 1\mathrm{d}u\big)\mathbb{E}[\|u_{t_{\mathrm{in}}}^{(i)}\|^2] + \int_{t_{\mathrm{in}}}^t \exp\big(-2\int_u^t \frac{1}{h_{T-s}} - 1\mathrm{d}s\big)\epsilon_{T-u}^{(i)}\mathrm{d}u.$$

When $t = T - \delta$ and $\delta \ll 1$, we have

$$\int_{t_{\mathrm{in}}}^{T-\delta} \exp\big(-2\int_u^{T-\delta} \frac{1}{h_{T-s}} - 1\mathrm{d}s\big)\epsilon_{T-u}^{(i)}\mathrm{d}u = \mathcal{O}(d\delta\log(1/\delta)).$$

Hence we prove that as $t \to T - \delta$, with high probability (tends to 1 as $\delta \to 0^+$), Y_t will enter some region dominated by one of the center $\mathcal{B}_{T-t}^{(i)}(\theta_{T-t})$ and then be pulled towards the center $a_{T-t}x_0^{(i)}$. Furthermore, quantitatively, the expected square-norm of $Y_{t-\delta} - a_\delta x_0^{(i)}$ is of order $\mathcal{O}(\delta)$.

For the reverse-time inference process $\tilde{Y}_t$, we utilize the same idea. Define

$$\tilde{r}_t^{(i)} = \tilde{Y}_t - m_{T-t}^{(i)}, \quad \tilde{u}_t^{(i)} \coloneqq (U_{T-t}^{(i)})^\intercal(\tilde{Y}_t - m_{T-t}^{(i)}).$$

The following the same estimations, we have

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}[\|\tilde{u}_t^{(i)}\|^2] \le -2(\frac{1}{h_{T-t}} - 1)\mathbb{E}[\|\tilde{u}_t^{(i)}\|^2] - \frac{4}{h_{T-t}}\langle\tilde{u}_t^{(i)}, (U_{T-t}^{(i)})^\intercal e_A(T-t, \tilde{Y}_t)\rangle + \tilde{\epsilon}_{T-t}^{(i)},$$

where $\tilde{\epsilon}_{T-t}^{(i)} = \mathcal{O}(1)$ as $t \to T^-$. Apply Young's inequality, we have $\frac{4}{h_{T-t}}\langle\tilde{u}_t^{(i)}, (U_{T-t}^{(i)})^\intercal e_A(T-t, \tilde{Y}_t)\rangle \le \frac{1}{h_{T-t}}\|\tilde{u}_t^{(i)}\|^2 + \frac{4}{h_{T-t}}\|(U_{T-t}^{(i)})^\intercal e_A(T-t, \tilde{Y}_t)\|^2$. Therefore,

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}[\|\tilde{u}_t^{(i)}\|^2] \le -(\frac{1}{h_{T-t}} - 2)\mathbb{E}[\|\tilde{u}_t^{(i)}\|^2] + \frac{4}{h_{T-t}}\|(U_{T-t}^{(i)})^\intercal e_A(T-t, \tilde{Y}_t)\|^2 + \tilde{\epsilon}_{T-t}^{(i)}.$$

Apply Gronwall's inequality and we get

$$\begin{aligned}\mathbb{E}[\|\tilde{u}_t^{(i)}\|^2] \le \exp\big(-\int_{\tilde{t}_{\mathrm{in}}}^t \frac{1}{h_{T-u}} - 2\mathrm{d}u\big)\mathbb{E}[\|\tilde{u}_{\tilde{t}_{\mathrm{in}}}^{(i)}\|^2] + \int_{\tilde{t}_{\mathrm{in}}}^t \exp\big(-\int_u^t \frac{1}{h_{T-s}} - 2\mathrm{d}s\big)\epsilon_{T-u}^{(i)}\mathrm{d}u \\ + \int_{\tilde{t}_{\mathrm{in}}}^t \exp\big(-\int_u^t \frac{1}{h_{T-s}} - 2\mathrm{d}s\big)\frac{4}{h_{T-u}}\|(U_{T-u}^{(i)})^\intercal e_A(T-u, \tilde{Y}_u)\|^2\mathrm{d}u.\end{aligned}$$

When $t = T - \delta$ and $\delta \ll 1$, we have

$$\begin{aligned}&\int_{\tilde{t}_{\mathrm{in}}}^{T-\delta} \exp\big(-\int_u^{T-\delta} \frac{1}{h_{T-s}} - 2\mathrm{d}s\big)\epsilon_{T-u}^{(i)}\mathrm{d}u = \mathcal{O}(\sqrt{\delta}), \\ &\int_{\tilde{t}_{\mathrm{in}}}^{T-\delta} \exp\big(-\int_u^{T-\delta} \frac{1}{h_{T-s}} - 2\mathrm{d}s\big)\frac{4}{h_{T-u}}\|(U_{T-u}^{(i)})^\intercal e_A(T-u, \tilde{Y}_u)\|^2\mathrm{d}u \\ =\ &\mathcal{O}(\sqrt{\delta}\int_{\tilde{t}_{\mathrm{in}}}^{T-\delta} h_{T-t}^{-\frac{3}{2}}\|(U_{T-u}^{(i)})^\intercal e_A(T-u, \tilde{Y}_u)\|^2\mathrm{d}u).\end{aligned}$$

□

Remark G.2 (Tangential distribution of the residual). To study the distribution of residual $\tilde{Y}_{T-\delta} - m_\delta^{(i)}$ along the tangent direction, we look at the SDE of $\|\tilde{u}_t^{(i)}\|^2$ and only look at the leading order drift as $t \to T^-$ and the diffusion term:

$$\begin{aligned}
\mathrm{d}\|\tilde{u}_t^{(i)}\|^2 &= -2(\frac{2}{h_{T-t}} - 1)\|\tilde{u}_t^{(i)}\|^2\mathrm{d}t + 2\langle u_t^{(i)}, \partial_t(U_{T-t}^{(i)})^\intercal(Y_t - m_{T-t}^{(i)})\rangle\mathrm{d}t \\
&\quad + 2\langle \tilde{u}_t^{(i)}, (U_{T-t}^{(i)})^\intercal\big(b_A(T-t,\tilde{Y}_t) - (\tilde{Y}_t + \frac{2}{h_{T-t}}(m_{T-t}^{(i)} - \tilde{Y}_t))\big)\rangle\mathrm{d}t \\
&\quad + 2d^*\mathrm{d}t + 2\sqrt{2}\langle \tilde{u}_t^{(i)}, (U_{T-t}^{(i)})^\intercal\mathrm{d}\tilde{B}_t\rangle \\
&= -\frac{4}{h_{T-t}}\|\tilde{u}_t^{(i)}\|^2\mathrm{d}t - \frac{4}{h_{T-t}}\langle \tilde{u}_t^{(i)}, (U_{T-t}^{(i)})^\intercal e_A(T-t,\tilde{Y}_t)\rangle\mathrm{d}t \\
&\quad + \mathcal{O}(1)\mathrm{d}t + 2\sqrt{2}\langle \tilde{u}_t^{(i)}, (U_{T-t}^{(i)})^\intercal\mathrm{d}\tilde{B}_t\rangle
\end{aligned}$$

which implies

$$\mathrm{d}\tilde{u}_t^{(i)} = \underbrace{-\frac{2}{h_{T-t}}\big(\tilde{u}_t^{(i)} + (U_{T-t}^{(i)})^\intercal e_A(T-t,\tilde{Y}_t)\big)\mathrm{d}t}_{\text{dominant drift}} + \sqrt{2}\mathrm{d}\tilde{B}_t + \mathcal{O}(1)\mathrm{d}t.$$

Notice that

$$\tilde{Y}_t = m_{T-t}^{(i)} + U_{T-t}^{(i)}\tilde{u}_t^{(i)} + \underbrace{(I_d - U_{T-t}^{(i)}(U_{T-t}^{(i)})^\intercal)(Y_t - m_{T-t}^{(i)})}_{\text{normal component}}$$

Assume the normal component is negligible, then we have

$$(U_{T-t}^{(i)})^\intercal e_A(T-t,\tilde{Y}_t) \approx (U_{T-t}^{(i)})^\intercal e_A(T-t, m_{T-t}^{(i)} + U_{T-t}^{(i)}\tilde{u}_t^{(i)}).$$

Then the approximate SDE for $\tilde{u}_t^{(i)}$ is given by

$$\mathrm{d}\tilde{u}_t^{(i)} = -\frac{2}{h_{T-t}}\big(\tilde{u}_t^{(i)} + (U_{T-t}^{(i)})^\intercal e_A(T-t, m_{T-t}^{(i)} + U_{T-t}^{(i)}\tilde{u}_t^{(i)})\big)\mathrm{d}t + \sqrt{2}\mathrm{d}\tilde{B}_t. \tag{19}$$

Lemma G.3. *Under Assumption 3.1, if $x \in \mathcal{B}_{T-t}^{(i)}(\eta_{T-t})$, then*

$$\|b(T-t,x) - \big(x + \frac{2}{h_{T-t}}(m_{T-t}^{(i)} - x)\big)\| \le 4Ra_{T-t}h_{T-t}^{-1}\theta_{T-t}.$$

Proof of Lemma G.3. According to Lemma D.7, we have

$$m(T-t,x) - m_{T-t}^{(i)} = \sum_{j\neq i}\mathrm{Softmax}(-\frac{\|x - m_{T-t}\|^2}{2h_{T-t}})_j a_{T-t}(x_0^{(i)} - x_0^{(i)}).$$

Under Assumption 3.1 and the definition of $\mathcal{B}_{T-t}^{(i)}(\eta_{T-t})$, we immediately have $\|m(T-t,x) - m_{T-t}^{(i)}\| \le 2Ra_{T-t}\theta_{T-t}$. Therefore,

$$\begin{aligned}
&\|b(T-t,x) - \big(x + \frac{2}{h_{T-t}}(m_{T-t}^{(i)} - x)\big)\| \\
&= \|x + \frac{2}{h_{T-t}}(m(T-t,x) - x) - \big(x + \frac{2}{h_{T-t}}(m_{T-t}^{(i)} - x)\big)\| \\
&= \frac{2}{h_{T-t}}\|m(T-t,x) - m_{T-t}^{(i)}\| \le 4Ra_{T-t}h_{T-t}^{-1}\theta_{T-t}.
\end{aligned}$$

□

Lemma G.4. *The local tangent frame $U_t^{(i)}$ satisfies that $\langle u, (\partial_t U_t^{(i)})^\intercal U_t^{(i)} u\rangle = 0$ for any $u \in \mathbb{R}^{d^*}$.*

Proof of Lemma G.4. Starting from $(U_t^{(i)})^\intercal U_t^{(i)} = I_{d^*}$ and take derivative wrt. t on both side and we get

$$(\partial_t U_t^{(i)})^\intercal U_t^{(i)} = -\big((\partial_t U_t^{(i)})^\intercal U_t^{(i)}\big)^\intercal.$$

Therefore, $(\partial_t U_t^{(i)})^\intercal U_t^{(i)}$ is skew-symmetric. Hence we proved (1). □

H. From Inference to Training

This section converts the error-bound terms in Theorems 3.6 and 3.7, which are expectations along the simulated inference path $\tilde{Y}_t$, into quantities that depend only on the exact reverse-time OU path $X_t^{\leftarrow}$. The key tool is a change of measure (Girsanov's theorem). A minor technical issue is that $\tilde{Y}_0 \sim \mathcal{N}(0, I_d)$ while $X_0^{\leftarrow} \sim p_T$. To isolate the initialization error, we introduce an auxiliary process $\hat{Y}_t$ that shares the transition kernel with $\tilde{Y}_t$ but starts from the correct initialization p_T.

Reverse-time processes. Let $e_A(t,x) \coloneqq m_A(t,x) - m(t,x)$. Consider the following reverse-time processes on $[0, T-\delta]$:

$$\mathrm{d}X_t^{\leftarrow} = \underbrace{(X_t^{\leftarrow} - \frac{2}{h_{T-t}} X_t^{\leftarrow} + \frac{2}{h_{T-t}} m(T-t, X_t^{\leftarrow}))}_{b(T-t, X_t^{\leftarrow})}\,\mathrm{d}t + \sqrt{2}\mathrm{d}B_t^{\leftarrow}, \quad X_0^{\leftarrow} \sim p_T,$$

$$\mathrm{d}\hat{Y}_t = \underbrace{(\hat{Y}_t - \frac{2}{h_{T-t}} \hat{Y}_t + \frac{2}{h_{T-t}} m_A(T-t, \hat{Y}_t))}_{\coloneqq b_A(T-t, \hat{Y}_t)}\,\mathrm{d}t + \sqrt{2}\mathrm{d}\tilde{B}_t, \quad \hat{Y}_0 \sim p_T,$$

$$\mathrm{d}\tilde{Y}_t = \underbrace{(\tilde{Y}_t - \frac{2}{h_{T-t}} \tilde{Y}_t + \frac{2}{h_{T-t}} m_A(T-t, \tilde{Y}_t))}_{\coloneqq b_A(T-t, \tilde{Y}_t)}\,\mathrm{d}t + \sqrt{2}\mathrm{d}\tilde{B}_t, \quad \tilde{Y}_0 \sim \mathcal{N}(0, I_d).$$

We denote the path measures of $X_t^{\leftarrow}$ and $\hat{Y}_t$ respectively by $\mathbb{P}$ and $\mathbb{Q}$. Since $X_0 \leftarrow$ and $\hat{Y}_0$ have the same initial distribution, we can apply Girsanov's theorem without evolving extra initial-density ratio factor. The remaining gap between $\tilde{Y}_t$ and $\hat{Y}_t$ is purely from initialization, and it will be controlled since p_T is close to $\mathcal{N}(0, I_d)$ when T is large.

Assumption H.1. We assume the following hold:

(0) Novikov's condition: $\exp\left(2\int_0^{T-\delta} \frac{\|e_A(T-s, X_t^{\leftarrow})\|^2}{h_{T-s}^2}\mathrm{d}s\right) < \infty$;

(1) Bound χ^2 along trajectory: $\sup_{t\in[0,T-\delta]} \chi^2(\mathbb{Q}_t|\mathbb{P}_t) \le C_\chi^2 = \mathcal{O}(1)$;

(2) Normal/tangent posterior mean error satisfies: for any $\dagger \in \{\perp, \|\}$, $\mathbb{E}[\|e_A^\dagger(T-t, X_t^{\leftarrow})\|^4]^{\frac{1}{2}} \le C_m \mathbb{E}[\|e_A^\dagger(T-t, X_t^{\leftarrow})\|^2]$ for some $C_m = \mathcal{O}(1)$;

(3) Uniform boundedness of the weighted posterior mean error:

$$\sup_{t\in[0,T-\delta]} \sup_{x\in\mathcal{T}_{T-t}(\rho_{T-t})} \frac{w(T-t)\|e_A(T-t,x)\|^2}{h_{T-t}^2} \le \frac{C_u}{T-\delta}$$

for some $C_u = \mathcal{O}(1)$ and decreasing weight $t \mapsto w(t)$.

Theorem H.2. *Under Assumption H.1, the normal/tangent-error floors in Theorems 3.6 and 3.7 can be estimated by the projected posterior mean matching loss in normal/tangent direction, i.e.,*

$$\textit{normal-error floor} \lesssim C_\delta^\perp \int_\delta^T \frac{w(t)\mathbb{E}[\|e_A^\perp(t, X_t)\|^2]}{h_t^2}\mathrm{d}t + C_\delta^\perp(\sqrt{d}+R)e^{-T} + d\delta^c, \tag{20}$$

$$\textit{tangent-error floor} \lesssim C_\delta^\| \int_\delta^T \frac{\mathbb{E}[w(t)\|e_A^\|(t, X_t)\|^2]}{h_t^2}\mathrm{d}t + C_\delta^\|(\sqrt{d}+R)e^{-T} + d\delta^{\frac{1}{2}}. \tag{21}$$

where $C_\delta^\perp \coloneqq \delta^c\big(1 \vee \frac{\delta^{1-c}}{w(\delta)}\big)$ *(*$c = \lim_{t\to\delta^+} h_t\beta_t$*) and* $C_\delta^\| = \sqrt{\delta}\big(1 \vee \frac{\sqrt{\delta}}{w(\delta)}\big)$.

Proof of Theorem H.2. Applying Girsanov's Theorem, define $\mathcal{L}_t \coloneqq \sqrt{2}\int_0^t \frac{e_A(T-s, X_s^{\leftarrow})}{h_{T-s}}\mathrm{d}B_s^{\leftarrow}$ and according to Assumption H.1-(0), we have $\mathcal{E}(\mathcal{L})_t \coloneqq \exp\left(\mathcal{L}_t - \frac{1}{2}[\mathcal{L}]_t\right)$ is a $\mathbb{P}$-martingale and

$$t \mapsto B_t^{\leftarrow} - \sqrt{2}\int_0^t \frac{e_A(T-s, X_s^{\leftarrow})}{h_{T-s}}\mathrm{d}s$$

is a Brownina motion under $\mathcal{E}(\mathcal{L})_{T-\delta}\mathbb{P}$ and $\mathbb{Q} = \mathcal{E}(\mathcal{L})_{T-\delta}\mathbb{P}$. Therefore, we can change the path measure from $\mathbb{Q}$ to $\mathbb{P}$, hence relating the inference bounds to the training error.

Recall that with $e_A(t,x) = m_A(t,x) - m(t,x)$ and $e_A^\perp(t,x) = P_t^\perp(x)e_A(t,x)$. When $\beta_t = c/h_t$ for some $c \in [1/2, 1)$, the normal-error term related to training in Theorem 3.6 is

$$\begin{aligned}
&\int_{\tilde{t}_{\text{in}}}^{T-\delta} \exp(-2\int_u^{T-\delta}(\beta_{T-s}-1)\mathrm{d}s)\frac{\mathbb{E}[\|e_A^\perp(T-u,\tilde{Y}_u)\|^2]}{h_{T-u}^2\beta_{T-u}}\mathrm{d}u \\
\lesssim\ & \delta^c\int_{\tilde{t}_{\text{in}}}^{T-\delta}\frac{h_{T-t}^{1-c}}{w(T-t)}\frac{w(T-t)\mathbb{E}[\|e_A^\perp(T-t,\tilde{Y}_t)\|^2]}{h_{T-t}^2}\mathrm{d}t \\
\lesssim\ & \delta^c\big(1\vee\frac{\delta^{1-c}}{w(\delta)}\big)\int_{\tilde{t}_{\text{in}}}^{T-\delta}\frac{w(T-t)\mathbb{E}[\|e_A^\perp(T-t,\tilde{Y}_t)\|^2]}{h_{T-t}^2}\mathrm{d}t,
\end{aligned}$$

WLOG, assume that $\tilde{t}_{\text{in}} = 0$. According to Girsanov's theorem,

$$\begin{aligned}
&\int_0^{T-\delta}\frac{\mathbb{E}[w(T-t)\|e_A^\perp(T-t,\tilde{Y}_t)\|^2]}{h_{T-t}^2}\mathrm{d}t \\
=&\int_0^{T-\delta}\frac{w(T-t)\mathbb{E}[\|e_A^\perp(T-t,\tilde{Y}_t)\|^2]-\|e_A^\perp(T-t,\hat{Y}_t)\|^2]}{h_{T-t}^2}\mathrm{d}t \\
&+\int_0^{T-\delta}\frac{w(T-t)\mathbb{E}[\mathcal{E}(\mathcal{L})_t\|e_A^\perp(T-t,X_t^{\leftarrow})\|^2]}{h_{T-t}^2}\mathrm{d}t.
\end{aligned}$$

Notice that

$$\begin{aligned}
\mathbb{E}[\mathcal{E}(\mathcal{L})_t\|e_A^\perp(T-t,X_t^{\leftarrow})\|^2] &= \mathbb{E}[\|e_A^\perp(T-t,X_t^{\leftarrow})\|^2] + \mathbb{E}[(\mathcal{E}(\mathcal{L})_t-1)\|e_A^\perp(T-t,X_t^{\leftarrow})\|^2] \\
&\le \mathbb{E}[\|e_A^\perp(T-t,X_t^{\leftarrow})\|^2] + \mathbb{E}[(\mathcal{E}(\mathcal{L})_t-1)^2]^{\frac{1}{2}}\mathbb{E}[\|e_A^\perp(T-t,X_t^{\leftarrow})\|^4]^{\frac{1}{2}} \\
&= \mathbb{E}[\|e_A^\perp(T-t,X_t^{\leftarrow})\|^2] + \sqrt{\chi^2(\mathbb{Q}_t|\mathbb{P}_t)}\mathbb{E}[\|e_A^\perp(T-t,X_t^{\leftarrow})\|^4]^{\frac{1}{2}}
\end{aligned}$$

Under Assumption H.1-(1)(2), we have

$$\int_0^{T-\delta}\frac{w(T-t)\mathbb{E}[\mathcal{E}(\mathcal{L})_t\|e_A^\perp(T-t,X_t^{\leftarrow})\|^2]}{h_{T-t}^2}\mathrm{d}t \le (1+C_\chi C_m)\int_0^{T-\delta}\frac{w(T-t)\mathbb{E}[\|e_A^\perp(T-t,X_t^{\leftarrow})\|^2]}{h_{T-t}^2}\mathrm{d}t.$$

Meanwhile, under Assumption H.1-(3), we have

$$\int_0^{T-\delta}\frac{w(T-t)\mathbb{E}[\|e_A^\perp(T-t,\tilde{Y}_t)\|^2]-\|e_A^\perp(T-t,\hat{Y}_t)\|^2]}{h_{T-t}^2}\mathrm{d}t \le C_u\mathrm{TV}(\mathcal{N}(0,I_d),p_T) \lesssim (\sqrt{d}+R)e^{-T}.$$

Therefore, the normal-error floor for aligning can be estimated as

$$\begin{aligned}
&\delta^c\big(1\vee\frac{\delta^{1-c}}{w(\delta)}\big)\int_0^{T-\delta}\frac{\mathbb{E}[\|e_A^\perp(T-u,\tilde{Y}_u)\|^2]}{h_{T-u}^2}\mathrm{d}u \\
\lesssim\ & \delta^c\big(1\vee\frac{\delta^{1-c}}{w(\delta)}\big)\int_0^{T-\delta}\frac{w(T-t)\mathbb{E}[\|e_A^\perp(T-t,X_t^{\leftarrow})\|^2]}{h_{T-t}^2}\mathrm{d}t + \delta^c\big(1\vee\frac{\delta^{1-c}}{w(\delta)}\big)(\sqrt{d}+R)e^{-T}.
\end{aligned}$$

Similarly, along the tangent direction, the tangent floor is of the form

$$\begin{aligned}
&\int_{\tilde{t}_{\text{in}}}^{T-\delta}\exp\big(-\int_u^{T-\delta}\frac{1}{h_{T-s}}-2\mathrm{d}s\big)\frac{4}{h_{T-u}}\|(U_{T-u}^{(i)})^\intercal e_A(T-u,\tilde{Y}_u)\|^2\mathrm{d}u \\
\lesssim\ & \sqrt{\delta}\big(1\vee\frac{\sqrt{\delta}}{w(\delta)}\big)\int_{\tilde{t}_{\text{in}}}^{T-\delta}\frac{w(T-u)\mathbb{E}[\|(U_{T-u}^{(i)})^\intercal e_A(T-u,\tilde{Y}_u)\|^2]}{h_{T-u}^2}\mathrm{d}u,
\end{aligned}$$

where the order estimation follows from the choice of $\beta_t = \Theta(1/h_t)$. When $\tilde{Y}_t \in \mathcal{T}_{T-t}(\rho_{T-t}) \cap \mathcal{B}^{(i)}(\theta_{T-t})$ and $t \to T-\delta$, we have $U^{(i)}_{T-t}(U^{(i)}_{T-t})^\intercal \approx P^{\|}_{T-t}(m^{(i)}_{T-t}) \approx P^{\|}_{T-t}(\tilde{Y}_t)$. Hence we can use the following tangent error to reflect the tangent-floor of the sliding phase:

$$\sqrt{\delta}\big(1 \vee \frac{\sqrt{\delta}}{w(\delta)}\big) \int_{\tilde{t}_{\text{in}}}^{T-\delta} \frac{w(T-t)\mathbb{E}[\|P^{\|}_{T-t}(\tilde{Y}_t)e_A(T-t,\tilde{Y}_t)\|^2]}{h^2_{T-t}}\mathrm{d}t$$
$$:= \sqrt{\delta}\big(1 \vee \frac{\sqrt{\delta}}{w(\delta)}\big) \int_{\tilde{t}_{\text{in}}}^{T-\delta} \frac{w(T-t)\mathbb{E}[\|e^{\|}_A(T-t,\tilde{Y}_t)\|^2]}{h^2_{T-t}}\mathrm{d}t.$$

Then following the same change of measure discussion, the tangent-error floor for sliding can be estimated as

$$\sqrt{\delta}\big(1 \vee \frac{\sqrt{\delta}}{w(\delta)}\big) \int_{0}^{T-\delta} \frac{w(T-u)\mathbb{E}[\|e^{\|}_A(T-u,\tilde{Y}_u)\|^2]}{h^2_{T-u}}\mathrm{d}u$$
$$\lesssim \sqrt{\delta}\big(1 \vee \frac{\sqrt{\delta}}{w(\delta)}\big) \int_{0}^{T-\delta} \frac{w(T-t)\mathbb{E}[\|e^{\|}_A(T-t,X^{\leftarrow}_t)\|^2]}{h^2_{T-t}}\mathrm{d}t + \sqrt{\delta}\big(1 \vee \frac{\sqrt{\delta}}{w(\delta)}\big)(\sqrt{d}+R)e^{-T}.$$

Last, the theorem follows from writing the reverse-time OU $X^{\leftarrow}_t$ into the forward-time OU X_t and adding the contraction terms depending on d, R in Theorems 3.6 and 3.7. □

I. Analysis of the Training Process

Recall from (7) that

$$A_{k+1} = A_k(I_p - 2\eta\tilde{U}) + 2\eta\tilde{V},$$

with $U_t = \mathbb{E}[\sigma_t(X_t(z))\sigma_t(X_t(z))^\intercal] \in \mathbb{R}^{p\times p}$ and $V_t = \mathbb{E}_z[a_t X_0 \sigma_t(X_t(z))^\intercal] \in \mathbb{R}^{d\times p}$

$$\tilde{U} = \int_\delta^T \frac{w(t)}{h_t^2}\frac{\mathbb{E}[\sigma_t(X_t(z))\sigma_t(X_t(z))^\intercal]}{p}\mathrm{d}t, \quad \tilde{V} = \int_\delta^T \frac{w(t)}{h_t^2}\frac{\mathbb{E}[a_t X_0\sigma_t(X_t(z))^\intercal]}{\sqrt{p}}\mathrm{d}t.$$

$\tilde{U}$ is the RFNN kernel matrix in (7) with rank $r \le p$ and eigen-decomposition $\tilde{U} = \sum_{j=1}^r \lambda_j u_j u_j^\intercal$ with $\{u_j\}_{j\in[r]}$ being a set of orthonormal vectors in $\mathbb{R}^p$ and $\lambda_1 > \lambda_2 > \cdots > \lambda_r > 0$.

We first provide initialization-dependent expressions for A_k and the equilibrium of gradient descent.

Apply the iteration recursively and we get

$$A_k = A_0(I_p - 2\eta\tilde{U})^k + 2\eta\tilde{V}\sum_{j=0}^{k-1}(I_p - 2\eta\tilde{U})^j.$$

If $2\eta\|\tilde{U}\| \le 1$, the gradient descent is stable. As $k \to \infty$,

$$A_k \to A_\infty = A_0(I_p - \tilde{U}\tilde{U}^+) + \tilde{V}\tilde{U}^+,$$

where $\tilde{U}^+$ is the pseudo-inverse of $\tilde{U}$.

Next, we present the error decomposition of $\mathcal{L}_{\text{MM}}(A_k)$ into architecture-driven error and optimization-driven error.

Proposition I.1. *Let $\{A_k\}$ be the matrix iterates from gradient descent (7) with learning rate $\eta < 2/\lambda_1$ and initialization A_0, then the $\mathcal{L}_{\text{MM}}$ can be decomposed as*

$$\mathcal{L}_{\text{MM}}(A_k) = \text{Err}_{arc} + \text{Err}_{train}(k), \quad \mathcal{L}_{\text{MM}}(A_\infty) = \text{Err}_{arc},$$

with

$$\text{Err}_{arc} = \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}[\|m(t,X_t(z)) - \frac{\tilde{V}\tilde{U}^+}{\sqrt{p}}\sigma_t(X_t(z))\|^2]\mathrm{d}t.$$
$$\text{Err}_{train}(k) = \sum_{i=1}^r \lambda_i(1-2\eta\lambda_i)^{2k}\|a_i\|^2,$$

and $a_j = (A_0 - \tilde{V}\tilde{U}^+)u_j \in \mathbb{R}^d$ for all $j \in [r]$.

Proof of Proposition I.1. Based on the dynamics of $\{A_k\}_{k\geq 0}$, we can study the dynamics of the DMM loss $\{\mathcal{L}_{\mathrm{DMM}}(A_k)\}_{k\geq 0}$. According to (5), we have

$$\begin{aligned}
\mathcal{L}_{\mathrm{DMM}}(A) &= \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}[\| - a_t X_0 + \frac{A}{\sqrt{p}}\sigma_t(X_t(z))\|^2]\mathrm{d}t \\
&= \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}[\| - a_t X_0\|^2]\mathrm{d}t + \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}[\|\frac{A}{\sqrt{p}}\sigma_t(X_t(z))\|^2]\mathrm{d}t \\
&\quad - 2\int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}[\langle a_t X_0, \frac{A}{\sqrt{p}}\sigma_t(X_t(z))\rangle]\mathrm{d}t \\
&= C + \mathrm{trace}(A\tilde{U}A^\intercal) - 2\mathrm{trace}(A\tilde{V}^\intercal),
\end{aligned} \tag{22}$$

where the constant $C = \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}[\|a_t X_0\|^2]\mathrm{d}t$ is independent to A. Notice that $\nabla_A \mathcal{L}_{\mathrm{DMM}}(A) = 2(A\tilde{U} - \tilde{V})$. Therefore, the equilibrium satisfy A^* satisfy $A^*\tilde{U} = \tilde{V}$ and they share the same DMM loss value $\mathcal{L}_{\mathrm{DMM}}(A^*)$. Next, we study the decay of DMM loss along gradient descent by tracking $\mathcal{L}_{\mathrm{DMM}}(A_k) - \mathcal{L}_{\mathrm{DMM}}(A^*)$. Define $\Delta A_k = A_k - A^*$, according to (22), we have

$$\begin{aligned}
\mathcal{L}_{\mathrm{DMM}}(A_k) &= C + \mathrm{trace}\big((A^* + \Delta A_k)\tilde{U}(A^* + \Delta A_k)^\intercal\big) - 2\mathrm{trace}\big((A^* + \Delta A_k)\tilde{V}^\intercal\big) \\
&= C + \mathrm{trace}(A^*\tilde{U}(A^*)^\intercal) - 2\mathrm{trace}(A^*\tilde{V}^\intercal) + \mathrm{trace}(\Delta A_k\tilde{U}\Delta A_k^\intercal) \\
&\quad + 2\cancel{\big(\mathrm{trace}(\Delta A_k\tilde{U}(A^*)^\intercal) - \mathrm{trace}(\Delta A_k\tilde{V}^\intercal)\big)} \\
&= \mathcal{L}_{\mathrm{DMM}}(A^*) + \mathrm{trace}(\Delta A_k\tilde{U}\Delta A_k^\intercal) \\
&= \mathcal{L}_{\mathrm{DMM}}(A^*) + \|\Delta A_k\tilde{U}^{\frac{1}{2}}\|_F^2,
\end{aligned}$$

where the last part in the second equation cancel due to the property of the equilibrium A^*. Meanwhile, according to the iteration formula, we can easily get that

$$\Delta A_{k+1} = A_{k+1} - A^* = \Delta A_k(I_p - 2\eta\tilde{U}) + 2\eta(A^*\tilde{U} + \tilde{V}) = \Delta A_k(I_p - 2\eta\tilde{U}),$$

where the last identity follows from the property of the equilibrium A^*. Combined with our equation of $\mathcal{L}_{\mathrm{DMM}}(A_k)$, we have

$$\mathcal{L}_{\mathrm{DMM}}(A_k) = \mathcal{L}_{\mathrm{DMM}}(A^*) + \|\Delta A_k\tilde{U}^{\frac{1}{2}}\|_F^2 = \mathcal{L}_{\mathrm{DMM}}(A^*) + \|\Delta A_0(I_p - 2\eta\tilde{U})^k\tilde{U}^{\frac{1}{2}}\|_F^2.$$

Therefore, based on the spectral information of $\tilde{U}$ and A_0, we can represent the DMM loss along gradient descent as follows

$$\mathcal{L}_{\mathrm{DMM}}(A_k) = \mathcal{L}_{\mathrm{DMM}}(A^*) + \sum_{i=1}^r \lambda_i(1 - 2\eta\lambda_i)^{2k}\|a_i\|^2. \tag{23}$$

In order to look at a loss with minimum strictly zero, we need to go back to the (non-denoising) posterior mean matching loss, i.e., $\mathcal{L}_{\mathrm{MM}}(A) = \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}[\| - m(t, X_t(z)) + \frac{A}{\sqrt{p}}\sigma_t(X_t(z))\|^2]\mathrm{d}t$. The DMM loss $\mathcal{L}_{\mathrm{DMM}}$ can be easily related to $\mathcal{L}_{\mathrm{MM}}$:

$$\begin{aligned}
\mathcal{L}_{\mathrm{DMM}}(A) &= \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}[\| - a_t X_0 + \frac{A}{\sqrt{p}}\sigma_t(X_t(z))\|^2]\mathrm{d}t \\
&= \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}[\| - m(t, X_t(z)) + \frac{A}{\sqrt{p}}\sigma_t(X_t(z))\|^2]\mathrm{d}t \\
&\quad + \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}[\| - a_t X_0 + m(t, X_t(z))\|^2]\mathrm{d}t \\
&\quad + 2\cancel{\int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}[\langle m(t, X_t(z)) - a_t X_0, \frac{A}{\sqrt{p}}\sigma_t(X_t(z))\rangle]\mathrm{d}t} \\
&= \mathcal{L}_{\mathrm{MM}}(A) + \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}[\| - a_t X_0 + m(t, X_t(z))\|^2]\mathrm{d}t,
\end{aligned}$$

where the cross term is canceled based on definition of $m(t,x)=\mathbb{E}[a_tX_0|X_t=x]$ and the tower property. Therefore, we can derive a decomposition of $\mathcal{L}_{\mathrm{MM}}(A_k)$ along the gradient descent dynamics:

$$\begin{aligned}
&\mathcal{L}_{\mathrm{MM}}(A_k) \qquad (24)\\
&=\mathcal{L}_{\mathrm{DMM}}(A_k)-\int_\delta^T\frac{w(t)}{h_t^2}\mathbb{E}[\|-a_tX_0+m(t,X_t(z))\|^2]\mathrm{d}t\\
&=\sum_{i=1}^r\lambda_i(1-2\eta\lambda_i)^{2k}\|a_i\|^2+\mathcal{L}_{\mathrm{DMM}}(A^*)-\int_\delta^T\frac{w(t)}{h_t^2}\mathbb{E}[\|-a_tX_0+m(t,X_t(z))\|^2]\mathrm{d}t\\
&=\sum_{i=1}^r\lambda_i(1-2\eta\lambda_i)^{2k}\|a_i\|^2+\int_\delta^T\frac{w(t)}{h_t^2}\mathbb{E}[\langle a_tX_0,a_tX_0-\frac{A^*}{\sqrt{p}}\sigma_t(X_t(z))\rangle]\mathrm{d}t\\
&\quad-\int_\delta^T\frac{w(t)}{h_t^2}\mathbb{E}[\|-a_tX_0+m(t,X_t(z))\|^2]\mathrm{d}t\\
&=\sum_{i=1}^r\lambda_i(1-2\eta\lambda_i)^{2k}\|a_i\|^2+\int_\delta^T\frac{w(t)}{h_t^2}\mathbb{E}[\langle m(t,X_t(z)),m(t,X_t(z))-\frac{A^*}{\sqrt{p}}\sigma_t(X_t(z))\rangle]\mathrm{d}t\\
&=\sum_{i=1}^r\lambda_i(1-2\eta\lambda_i)^{2k}\|a_i\|^2+\int_\delta^T\frac{w(t)}{h_t^2}\mathbb{E}[\|m(t,X_t(z))-\frac{A^*}{\sqrt{p}}\sigma_t(X_t(z))\|^2]\mathrm{d}t, \qquad (25)
\end{aligned}$$

where the second last property follows from the tower property and the last identity follows from the property of A^* and the tower property. Therefore, Equation (24) decompose the posterior mean error into two parts which induce different type of implicit bias.

(1) Architecture Implicit Bias: the term $\int_\delta^T\frac{w(t)}{h_t^2}\mathbb{E}[\|m(t,X_t(z))-\frac{A^*}{\sqrt{p}}\sigma_t(X_t(z))\|^2]\mathrm{d}t$ orients from the inadequate representation ability of the RFNN with finite p. Even though the equilibrium A^* can be any matrix of the form $A_0(I_p-\tilde{U}\tilde{U}^+)+\tilde{V}\tilde{U}^+$ (depending on initialization), the value of the architecture implicit bias stays the same. Therefore, we can pick a special $A^*=\tilde{V}\tilde{U}^+$ to represent it, i.e.,

$$\mathrm{Err}_{arc}=\int_\delta^T\frac{w(t)}{h_t^2}\mathbb{E}[\|m(t,X_t(z))-\frac{\tilde{V}\tilde{U}^+}{\sqrt{p}}\sigma_t(X_t(z))\|^2]\mathrm{d}t.$$

(2) Training Dynamical Implicit Bias: the term $\sum_{i=1}^r\lambda_i(1-2\eta\lambda_i)^{2k}\|a_i\|^2$ is produced by running the optimization algorithm for finite time. If we can ideally run the gradient descent for infinite many step, this bias will vanish. We represent the training dynamical bias as follows

$$\mathrm{Err}_{train}(k)=\sum_{i=1}^r\lambda_i(1-2\eta\lambda_i)^{2k}\|a_i\|^2.$$

□

Based on Proposition I.1, we can proceed to prove Theorem 4.2.

Proof of Theorem 4.2. The decomposition of training dynamical/architecture implicit bias requires to repeat our analysis to the total DMM loss.

$$\begin{aligned}
\mathcal{L}^\perp_{\mathrm{DMM}}(A)&\coloneqq\int_\delta^T\frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^\perp(X_t(z))\big(-a_tX_0+\frac{A}{\sqrt{p}}\sigma_t(X_t(z))\big)\|^2]\mathrm{d}t\\
&=\underbrace{\int_\delta^T\frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^\perp(X_t(z))(-a_tX_0)\|^2]\mathrm{d}t}_{:=C^\perp}+\underbrace{\int_\delta^T\frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^\perp(X_t(z))(\frac{A}{\sqrt{p}}\sigma_t(X_t(z)))\|^2]\mathrm{d}t}_{\text{quadratic in }A}\\
&\quad\underbrace{-2\int_\delta^T\frac{w(t)}{h_t^2}\mathbb{E}[\langle P_t^\perp(X_t(z))(a_tX_0),\frac{A}{\sqrt{p}}\sigma_t(X_t(z))\rangle]\mathrm{d}t}_{\text{linear in }A}.
\end{aligned}$$

Then we have

$$\begin{aligned}
\mathcal{L}^{\perp}_{\mathrm{DMM}}(A_k) &= \mathcal{L}^{\perp}_{\mathrm{DMM}}(A^* + \Delta A_k) \\
&= C^{\perp} + \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^{\perp}(X_t(z))(\frac{A^* + \Delta A_k}{\sqrt{p}}\sigma_t(X_t(z)))\|^2]\mathrm{d}t \\
&\quad - 2\int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\langle P_t^{\perp}(X_t(z))(a_t X_0), \frac{A^* + \Delta A_k}{\sqrt{p}}\sigma_t(X_t(z))\rangle]\mathrm{d}t \\
&= \mathcal{L}^{\perp}_{\mathrm{DMM}}(A^*) + \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^{\perp}(X_t(z))(\frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z)))\|^2]\mathrm{d}t \\
&\quad + 2\int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\langle P_t^{\perp}(X_t(z))(\frac{A^*}{\sqrt{p}}\sigma_t(X_t(z)) - a_t X_0), \frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z))\rangle]\mathrm{d}t.
\end{aligned}$$

Therefore, the posterior mean loss is

$$\begin{aligned}
\mathcal{L}^{\perp}_{\mathrm{MM}}(A_k) &= \mathcal{L}^{\perp}_{\mathrm{DMM}}(A_k) - \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^{\perp}(X_t(z))\big(-a_t X_0 + m(t, X_t(z))\big)\|^2]\mathrm{d}t \\
&= \mathcal{L}^{\perp}_{\mathrm{DMM}}(A^*) + \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^{\perp}(X_t(z))(\frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z)))\|^2]\mathrm{d}t \\
&\quad + 2\int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\langle P_t^{\perp}(X_t(z))(\frac{A^*}{\sqrt{p}}\sigma_t(X_t(z)) - a_t X_0), \frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z))\rangle]\mathrm{d}t \\
&\quad - \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^{\perp}(X_t(z))\big(-a_t X_0 + m(t, X_t(z))\big)\|^2]\mathrm{d}t \\
&= \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^{\perp}(X_t(z))\big(m(t, X_t(z)) - \frac{A^*}{\sqrt{p}}\sigma_t(X_t(z))\big)\|^2]\mathrm{d}t \\
&\quad + \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^{\perp}(X_t(z))(\frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z)))\|^2]\mathrm{d}t \\
&\quad + 2\int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\langle P_t^{\perp}(X_t(z))(\frac{A^*}{\sqrt{p}}\sigma_t(X_t(z)) - m(t, X_t(z)), \frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z))\rangle]\mathrm{d}t.
\end{aligned}$$

Similarly, along the tangent directions, we have

$$\begin{aligned}
\mathcal{L}^{\|}_{\mathrm{MM}}(A_k) &= \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^{\|}(X_t(z))\big(m(t, X_t(z)) - \frac{A^*}{\sqrt{p}}\sigma_t(X_t(z))\big)\|^2]\mathrm{d}t \\
&\quad + \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^{\|}(X_t(z))(\frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z)))\|^2]\mathrm{d}t \\
&\quad + 2\int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\langle P_t^{\|}(X_t(z))(\frac{A^*}{\sqrt{p}}\sigma_t(X_t(z)) - m(t, X_t(z)), \frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z))\rangle]\mathrm{d}t.
\end{aligned}$$

Then along the normal and tangent directions, the architecture implicit bias are

$$\begin{aligned}
\mathrm{Err}^{\perp}_{arc} &= \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^{\perp}(X_t(z))\big(m(t, X_t(z)) - \frac{A^*}{\sqrt{p}}\sigma_t(X_t(z))\big)\|^2]\mathrm{d}t, \\
\mathrm{Err}^{\|}_{arc} &= \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^{\|}(X_t(z))\big(m(t, X_t(z)) - \frac{A^*}{\sqrt{p}}\sigma_t(X_t(z))\big)\|^2]\mathrm{d}t.
\end{aligned}$$

The training dynamical implicit bias along the tangent directions are

$$\begin{aligned}
\mathrm{Err}^{\perp}_{train}(k) &= \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^{\perp}(X_t(z))(\frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z)))\|^2]\mathrm{d}t \\
&\quad + 2\int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\langle P_t^{\perp}(X_t(z))(\frac{A^*}{\sqrt{p}}\sigma_t(X_t(z)) - m(t,X_t(z)), \frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z)))\rangle]\mathrm{d}t, \\
\mathrm{Err}^{\|}_{train}(k) &= \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^{\|}(X_t(z))(\frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z)))\|^2]\mathrm{d}t \\
&\quad + 2\int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\langle P_t^{\|}(X_t(z))(\frac{A^*}{\sqrt{p}}\sigma_t(X_t(z)) - m(t,X_t(z)), \frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z)))\rangle]\mathrm{d}t.
\end{aligned}$$

Again, since $\Delta A_k = \Delta A_0(1-2\eta\tilde{U})^k$, we have $\frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z)) = \sum_{j=1}^r (1-2\eta\lambda_j)^k \frac{u_j^\intercal \sigma_t(X_t(z))}{\sqrt{p}} a_j \coloneqq \sum_{j=1}^r (1-2\eta\lambda_j)^k \sigma_{t,u_j} a_j$. Then we have

$$\begin{aligned}
&\int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\|P_t^{\perp}(X_t(z))(\frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z)))\|^2]\mathrm{d}t \\
=&\sum_{j,l=1}^{r}(1-2\eta\lambda_j)^k(1-2\eta\lambda_l)^k a_j^\intercal \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\sigma_{t,u_j}\sigma_{t,u_l}P_t^{\perp}(X_t(z))]a_l \\
\coloneqq&\sum_{j,l=1}^{r}(1-2\eta\lambda_j)^k(1-2\eta\lambda_l)^k a_j^\intercal P_{jl}^{\perp} a_l \\
&2\int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\langle P_t^{\perp}(X_t(z))(\frac{A^*}{\sqrt{p}}\sigma_t(X_t(z)) - m(t,X_t(z)), \frac{\Delta A_k}{\sqrt{p}}\sigma_t(X_t(z)))\rangle]\mathrm{d}t \\
=&2\sum_{j=1}^{r}(1-2\eta\lambda_j)^k a_j^\intercal \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\sigma_{t,u_j}P_t^{\perp}(X_t(z))(\frac{A^*}{\sqrt{p}}\sigma_t(X_t(z)) - m(t,X_t(z))]\mathrm{d}t \\
\coloneqq&2\sum_{j=1}^{r}(1-2\eta\lambda_j)^k a_j^\intercal b_j^{\perp}.
\end{aligned}$$

We can work on the tangent directions similarly. Therefore, we have

$$\begin{aligned}
\mathrm{Err}^{\perp}_{train}(k) &= \sum_{j,l=1}^{r}(1-2\eta\lambda_j)^k(1-2\eta\lambda_l)^k a_j^\intercal P_{jl}^{\perp} a_l + 2\sum_{j=1}^{r}(1-2\eta\lambda_j)^k a_j^\intercal b_j^{\perp}, \\
\mathrm{Err}^{\|}_{train}(k) &= \sum_{j,l=1}^{r}(1-2\eta\lambda_j)^k(1-2\eta\lambda_l)^k a_j^\intercal P_{jl}^{\|} a_l + 2\sum_{j=1}^{r}(1-2\eta\lambda_j)^k a_j^\intercal b_j^{\|},
\end{aligned}$$

where

$$\begin{aligned}
P_{jl}^{\perp} &= \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\sigma_{t,u_j}\sigma_{t,u_l}P_t^{\perp}(X_t(z))]\mathrm{d}t, \quad b_j^{\perp} = \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\sigma_{t,u_j}P_t^{\perp}(X_t(z))(\frac{A^*}{\sqrt{p}}\sigma_t(X_t(z)) - m(t,X_t(z))]\mathrm{d}t, \\
P_{jl}^{\|} &= \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\sigma_{t,u_j}\sigma_{t,u_l}P_t^{\|}(X_t(z))]\mathrm{d}t, \quad b_j^{\|} = \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\sigma_{t,u_j}P_t^{\|}(X_t(z))(\frac{A^*}{\sqrt{p}}\sigma_t(X_t(z)) - m(t,X_t(z))]\mathrm{d}t \quad (26)
\end{aligned}$$

and recall $\sigma_{t,u_j} = \frac{u_j^\intercal \sigma_t(X_t(z))}{\sqrt{p}}$ for all $j \in [r]$. It is worth mentioning that

$$P_{jl}^{\perp} + P_{jl}^{\|} = \int_{\delta}^{T} \frac{w(t)}{h_t^2}\mathbb{E}[\sigma_{t,u_j}\sigma_{t,u_l}]\mathrm{d}t = \lambda_j 1_{j=l}, \quad b_j^{\perp} + b_l^{\|} = 0.$$

Hence $\mathrm{Err}_{train}(k) = \mathrm{Err}^{\perp}_{train}(k) + \mathrm{Err}^{\|}_{train}(k)$. □

J. Experimental Details

In this section, we list the specific model architectures and hyperparameters that were not included in the main text.

J.1. Two Points in 2D Plane

Model Architectures Our RFNN model strictly follows the architecture defined in Section 2, with a model width of $p = 2000$ and a temporal embedding dimension of $K_t = 128$.

Training and Loss Computation We optimize the model via gradient descent for 10^7 epochs with a learning rate of 5×10^{-4}. The gradient descent is performed using the full-batch SGD optimizer in PyTorch and loss function (5). We numerically evaluate the integral in the loss function (5) using the trapezoidal rule with 2,000 discretization points.

SDE and Sampling Details We model the diffusion process using a stochastic differential equation (SDE) with a drift coefficient of $f(x,t) = -x$ and a constant diffusion coefficient of $g(t) = \sqrt{2}$. For generation, we employ the Euler-Maruyama sampler with $N = 1000$ discrete time steps. The time schedule follows a geometric progression decaying from $T = 10$ down to $\delta = t_{min} = 10^{-3}$. Specifically, the time points are defined as $t_i = T \cdot (t_{min}/T)^{i/N}$ for $i = 0, \ldots, N$.

J.2. More Points in 2D Plane

We use the RFNN model with the same architecture, training, and sampling settings as described in Section J.1. The only differences are the training epochs: for the four-point case (Figure 2), we train for 5,000,000 epochs with $lr = 0.0005$; for the three-point case (Figure 3), we train for 3,000,000 epochs with $lr = 0.0005$.

J.3. MNIST

Model Architectures We utilize Multi-Layer Perceptrons (MLPs) for all components of our framework.

- **VAE Architecture:** The encoder maps the flattened input image ($\mathbb{R}^{784}$) to the latent space through two hidden layers with 400 and 200 units, respectively. It uses ReLU activations and outputs the mean and log-variance parameters. The decoder mirrors this structure (latent $\to$ 200 $\to$ 400 $\to$ 784) and uses a Sigmoid activation at the final layer to ensure pixel values lie within $[0, 1]$.
- **Latent Score Network:** The score model is a time-conditioned MLP. The time step t is first mapped to a 128-dimensional feature vector using Sinusoidal Embedding. This embedding is concatenated with the input vector and passed through three fully connected layers (sizes: input $\to$ 256 $\to$ 256 $\to$ latent dimension) with ReLU activations to estimate the score function.

Training Configuration All models are trained on a single GPU using the Adam optimizer with a batch size of 1024.

- **VAE Training:** The VAE is trained for 100 epochs with a learning rate of 10^{-3}. It minimizes the standard Evidence Lower Bound loss.
- **Score Model Training:** The latent diffusion model is trained for 300 epochs with a learning rate of 10^{-4}. The loss function follows the denoising score matching objective (5). The weight schedule in the loss function is set to a constant $w(t) = 1$.

SDE and Sampling Details We model the diffusion process using a stochastic differential equation (SDE) with a drift coefficient of $f(x,t) = -x$ and a constant diffusion coefficient of $g(t) = \sqrt{2}$. For generation, we employ the Euler-Maruyama sampler with $N = 1000$ discrete time steps. The time schedule follows a geometric progression decaying from $T = 10$ down to $\delta = t_{min} = 10^{-3}$. Specifically, the time points are defined as $t_i = T \cdot (t_{min}/T)^{i/N}$ for $i = 0, \ldots, N$.

J.4. Projection onto Log-density Ridge Sets

During the reverse sampling process, given a sample $x_0 \in \mathbb{R}^d$ generated by the diffusion model at a specific time step, our objective is to find its exact projection y^* onto the ridge manifold $\mathcal{R}_{d^*}(p; \beta)$. This naturally formulates as a constrained

non-linear optimization problem:

$$\begin{aligned} \min_{y\in\mathbb{R}^d} \quad & \frac{1}{2}|y-x_0|^2 \\ \text{s.t.} \quad & F(y) \coloneqq E(y)^\intercal \nabla \log p(y) = 0, \\ & \lambda_{d+1}(y) \le -\beta \end{aligned} \tag{27}$$

where $F(y) \in \mathbb{R}^{d-d^*}$ represents the residual projection of the score function onto the normal space. To solve this projection problem, we introduce the Lagrange multiplier $\nu \in \mathbb{R}^{d-d^*}$ for the equality constraint $F(y) = 0$. The unconstrained Lagrangian function is given by:

$$\mathcal{L}(y,\nu) = \frac{1}{2}|y-x_0|^2 + \nu^\intercal F(y) \tag{28}$$

According to the KKT conditions, the exact projection point must satisfy the following stationarity conditions:

$$\begin{cases} \nabla_y \mathcal{L}(y,\nu) = (y-x_0) + J(y)^\intercal \nu = 0 \\ F(y) = 0 \end{cases} \tag{29}$$

where $J(y) \coloneqq \nabla_y F(y) \in \mathbb{R}^{(d-d^*)\times d}$ is the Jacobian matrix of the normal residual. In high-dimensional spaces, approximating $J(y)$ via finite differences introduces severe numerical instability. Instead, we construct the Jacobian analytically. Applying the product rule to $F(y)$, we obtain:

$$J(y) = E(y)^\intercal \nabla^2 \log p(y) + \big(\nabla_y E(y)\big)^\intercal \nabla \log p(y) \tag{30}$$

Let $H(y) \coloneqq \nabla^2 \log p(y)$ denote the Hessian matrix. Since the column vectors of $E(y)$ are precisely the eigenvectors of $H(y)$ spanning the normal space, the first term simplifies directly to $\Lambda_\perp(y)E(y)^\intercal$, where $\Lambda_\perp(y) = \mathrm{diag}(\lambda_{d^*+1},\dots,\lambda_d)$.For the second term, which involves the derivative of the invariant subspace, we use matrix perturbation theory. Let $T(y) \in \mathbb{R}^{d\times d^*}$ be the orthogonal basis of the tangent space, and $\Lambda_\parallel(y)$ be the corresponding diagonal matrix of the tangent eigenvalues. The normal basis $E(y)$ satisfies the eigenvalue equation $H(y)E(y) = E(y)\Lambda_\perp(y)$. Differentiating both sides with respect to the coordinate component $y^{(\ell)}$ yields:

$$\frac{\partial H(y)}{\partial y^{(\ell)}}E(y) + H(y)\frac{\partial E(y)}{\partial y^{(\ell)}} = \frac{\partial E(y)}{\partial y^{(\ell)}}\Lambda_\perp(y) + E(y)\frac{\partial \Lambda_\perp(y)}{\partial y^{(\ell)}}$$

To isolate the variation of the normal space that alters the manifold's geometry—namely, its "tilt" towards the tangent space—we pre-multiply both sides by $T(y)^\intercal$. Utilizing the orthogonality $T(y)^\intercal E(y) = 0$ and the symmetry $T(y)^\intercal H(y) = \Lambda_\parallel(y)T(y)^\intercal$, the equation simplifies to a standard Sylvester equation:

$$\Lambda_\parallel(y)\left(T(y)^\intercal \frac{\partial E(y)}{\partial y^{(\ell)}}\right) - \left(T(y)^\intercal \frac{\partial E(y)}{\partial y^{(\ell)}}\right)\Lambda_\perp(y) = -T(y)^\intercal \frac{\partial H(y)}{\partial y^{(\ell)}}E(y)$$

Given the strict eigengap $\lambda_{d^*}(y) > \lambda_{d^*+1}(y)$, the spectra of $\Lambda_\parallel(y)$ and $\Lambda_\perp(y)$ are disjoint, guaranteeing a unique solution. Using the Kronecker sum $\oplus$, the projection of the derivative onto the tangent space can be solved analytically. By mapping this tangent projection back to the ambient space, the correction term induced by the partial derivative of $E(y)$ can be explicitly expressed as:

$$\frac{\partial E(y)}{\partial y^{(\ell)}} = -T(y)\big(\Lambda_\parallel(y) \oplus (-\Lambda_\perp(y))\big)^{-1}T(y)^\intercal \frac{\partial H(y)}{\partial y^{(\ell)}}E(y)$$

By aggregating this correction term across all dimensions $\ell \in \{1,\dots,d\}$, we compute the exact Jacobian matrix $J(y)$. This analytical construction effectively avoids the systematic biases inherent in second-order root-finding processes when tracking the "tilt" of the normal space.

Because $J(y)$ can become ill-conditioned when the initial point is far from the manifold, we employ a Levenberg-Marquardt (LM) approach with a damping parameter $\sigma > 0$ to solve the KKT system. At the k-th iteration, given the current point y_k, we linearize the constraint as $F(y_k + s) \approx F(y_k) + J(y_k)s$. We then solve the following regularized linear system to obtain the dual variable ν and the primal step s:

$$\begin{aligned} \big(J(y_k)J(y_k)^\intercal\big)\nu &= (1+\sigma)F(y_k) - J(y_k)(y_k - x_0) \\ s &= -\frac{(y_k - x_0) + J(y_k)^\intercal \nu}{1+\sigma} \end{aligned} \tag{31}$$

To ensure that the iteration sequence strictly converges to a point within the valid ridge manifold boundaries (i.e., satisfying the strict curvature condition $\lambda_{d^*+1}(y) \leq -\beta$ defined in Definition 3.2), we design a merit function with boundary penalties to guide the line search:

$$M(y) = \frac{1}{2}|y - x_0|^2 + \frac{c}{2}|F(y)|^2 + \gamma \max\left(0, \lambda_{d^*+1}(y) + \beta\right)^2 \tag{32}$$

where $c > 0$ is the penalty weight for the equality constraint, and $\gamma > 0$ is the penalty factor for curvature violations. During each iteration, if the trial step $y_{\text{trial}} = y_k + s$ yields a decrease in the merit function ($M(y_{\text{trial}}) < M(y_k)$), the step is accepted, and the damping parameter σ is reduced. Otherwise, the step is rejected, σ is increased, and the LM system is solved again. This procedure is repeated until the convergence criteria $\|F(y)\| \leq \epsilon_{\text{tol}}$ and $\lambda_{d^*+1}(y) \leq -\beta$ are simultaneously met.

In our experiments, to find the exact projection of generated samples onto the log-density ridge manifold during the reverse sampling phase, we set the intrinsic manifold dimension to $d^* = 3, 5, 8, 12$ and enforce a strict normal curvature margin of $\beta = 10^{-3}$. For solving the KKT system, we employ Levenberg-Marquardt optimization algorithm with a maximum of 30 iterations and a convergence tolerance of $\epsilon_{\text{tol}} = 10^{-6}$. A diagonal perturbation of $\epsilon_{\text{reg}} = 10^{-6}$ is introduced during the Jacobian inversion to ensure numerical stability, and the penalty weights for both the equality constraint and curvature violation in the merit function are set to $c = \gamma = 100.0$. Throughout the 1000-step diffusion generative trajectory, we dynamically monitor the geometric distance to the manifold every 100 time steps using 200 samples.

Algorithm 1 Projection onto Log-density Ridge Sets via Regularized LM

Input: Initial sample $x_0 \in \mathbb{R}^d$, score function $\nabla \log p(\cdot)$, intrinsic dimension d^*.
Parameters: Curvature margin $\beta = 10^{-3}$, tolerance $\epsilon_{\text{tol}} = 10^{-6}$, max iterations $N_{\max} = 30$, initial damping $\sigma = 10^{-3}$, KKT regularization $\epsilon_{\text{reg}} = 10^{-6}$, merit weights $c = 100, \gamma = 100$.
Initialize: $y \leftarrow x_0$
for $k = 0$ **to** $N_{\max} - 1$ **do**
  Compute score $g \leftarrow \nabla \log p(y)$ and Hessian $H \leftarrow \nabla^2 \log p(y)$
  Compute eigendecomposition of H to yield tangent basis $T \in \mathbb{R}^{d \times d^*}$, normal basis $E \in \mathbb{R}^{d \times (d-d^*)}$, and eigenvalue matrices $\Lambda_{\|}, \Lambda_{\perp}$
  Evaluate normal residual $F(y) \leftarrow E^{\intercal} g$
  if $\|F(y)\| \le \epsilon_{\text{tol}}$ **and** $\lambda_{d^*+1} \le -\beta$ **then**
    return y *Converged to exact ridge manifold*
  end if
  Construct analytical Jacobian $J(y) \in \mathbb{R}^{(d-d^*) \times d}$ via matrix perturbation
  for $\ell = 1$ **to** d **do**
    Compute normal basis derivative: $\frac{\partial E}{\partial y^{(\ell)}} \leftarrow -T\big(\Lambda_{\|} \oplus (-\Lambda_{\perp})\big)^{-1} T^{\intercal} \frac{\partial H}{\partial y^{(\ell)}} E$
  end for
  Assemble Jacobian via column aggregation: $J(y) \leftarrow \Lambda_{\perp} E^{\intercal} + \left[\left(\frac{\partial E}{\partial y^{(1)}} \right)^{\intercal} g, \ldots, \left(\frac{\partial E}{\partial y^{(d)}} \right)^{\intercal} g \right]$
  Solve the regularized KKT system for dual variable ν and primal step s
  Update dual variable: $\nu \leftarrow (J(y)J(y)^{\intercal} + \epsilon_{\text{reg}} I)^{-1} \big((1+\sigma) F(y) - J(y)(y - x_0) \big)$
  Compute primal step: $s \leftarrow -\frac{1}{1+\sigma} \big((y - x_0) + J(y)^{\intercal} \nu \big)$
  Define trial step: $y_{\text{trial}} \leftarrow y + s$
  Evaluate merit function to ensure constraints and boundary penalties
  Evaluate trial merit: $M(z) = \frac{1}{2}\|z - x_0\|^2 + \frac{c}{2}\|F(z)\|^2 + \gamma \max\big(0, \lambda_{d^*+1}(z) + \beta\big)^2$
  if $M(y_{\text{trial}}) < M(y)$ **then**
    Accept step and decrease damping: $y \leftarrow y_{\text{trial}}, \sigma \leftarrow \max(\sigma/2, \epsilon_{\text{reg}})$
  else
    Reject step and increase damping: $\sigma \leftarrow 10 \cdot \sigma$
  end if
end for
return y *Return best approximation if max iterations reached*

J.5. Stability

In the tables below, we track the dynamic geometric relationship between the generated samples and the local ridge manifold during the reverse sampling process. The columns are defined as follows: **Step** and t represent the iteration step and the corresponding noise scale. **Mean Dist.** is the average geometric projection distance from the samples to the ridge manifold. **Curv. OK** counts the samples that strictly meet the negative curvature condition ($\lambda_{d^*+1} \leq -\beta$). **Mean Resid.** shows the initial normal score residual norm ($\|F(x_t)\|$) before projection. **Mean Gap** is the average eigengap between the tangent and normal spaces, which measures how well the manifold structure is separated. **Failures** counts how many times the Levenberg-Marquardt projection algorithm failed to converge in 30 steps.

Importantly, the Reliable column counts samples that successfully converged with an eigengap strictly greater than zero ($\lambda_{d^*} > \lambda_{d^*+1}$). However, we must note a numerical behavior in the very early stages of generation (e.g., Steps 0 and 100). Although these samples are counted as "Reliable" because their gap is technically > 0, their actual eigengaps are extremely small (typically $< 10^{-6}$). Physically, this means the local geometry is very close to random isotropic Gaussian noise, causing the tangent and normal spaces to blend together. Because a clear manifold structure has not yet formed in this high-noise phase, the algorithm's numerical stability is weak. Therefore, the projection distances measured at these earliest steps have limited geometric meaning. Based on this observation , we deliberately omit the initial high-noise stages at $t = 10.0$ and $t = 4.67$ (i.e., Steps 0 and 100) in Figure 8(a).

Step	t	Mean Dist.	Reliable	Curv. OK	Mean Resid.	Mean Gap	Failures
0	10.00000	5.373256	200	200	5.373264	0.000000	0
100	4.67624	3.108178	200	200	3.108809	0.000005	0
200	2.18672	3.038782	200	200	3.072826	0.000740	0
300	1.02257	2.805413	200	200	3.141728	0.014021	0
400	0.47818	2.339501	200	200	3.570831	0.081940	0
500	0.22361	1.990165	180	180	4.398019	0.275265	0
600	0.10456	1.927771	68	80	6.084632	1.476575	12
700	0.04890	1.191260	128	128	10.155400	5.755606	0
800	0.02287	0.971192	196	196	19.608429	4.012766	0
900	0.01069	0.919455	200	200	40.914448	1.886402	0
1000	0.00500	0.911901	200	200	86.183301	0.003698	0

Table 1. Dynamic manifold distance monitoring data during the reverse sampling process of the diffusion model (intrinsic dimension $d^* = 3$).

Step	t	Mean Dist.	Reliable	Curv. OK	Mean Resid.	Mean Gap	Failures
0	10.00000	5.130307	200	200	5.130310	0.000000	0
100	4.67624	2.989573	200	200	2.989411	0.000007	0
200	2.18672	2.882148	200	200	2.919268	0.001082	0
300	1.02257	2.724726	200	200	3.124859	0.013572	0
400	0.47818	2.207066	200	200	3.423936	0.058334	0
500	0.22361	1.710056	200	200	4.212194	0.196339	0
600	0.10456	1.407467	180	180	5.857416	0.852296	0
700	0.04890	1.058827	192	192	9.560239	2.028122	0
800	0.02287	0.799892	200	200	16.567018	0.794553	0
900	0.01069	0.741769	200	200	32.952072	0.029825	0
1000	0.00500	0.664260	200	200	65.937224	0.001428	0

Table 2. Dynamic manifold distance monitoring data during the reverse sampling process of the diffusion model (intrinsic dimension $d^* = 5$).

Step	t	Mean Dist.	Reliable	Curv. OK	Mean Resid.	Mean Gap	Failures
0	10.00000	4.770034	200	200	4.770048	0.000000	0
100	4.67624	2.750176	200	200	2.750356	0.000002	0
200	2.18672	2.737994	200	200	2.785426	0.000339	0
300	1.02257	2.592569	200	200	3.022125	0.008972	0
400	0.47818	2.179726	200	200	3.496142	0.052296	0
500	0.22361	1.609191	200	200	4.321711	0.142764	0
600	0.10456	1.086982	200	200	5.644727	0.440631	0
700	0.04890	0.790139	200	200	8.168482	0.438126	0
800	0.02287	0.602685	200	200	13.046494	0.033540	0
900	0.01069	0.446253	200	200	20.609062	0.003741	0
1000	0.00500	0.394242	200	200	37.758054	0.000013	0

Table 3. Dynamic manifold distance monitoring data during the reverse sampling process of the diffusion model (intrinsic dimension $d^* = 8$).

Step	t	Mean Dist.	Reliable	Curv. OK	Mean Resid.	Mean Gap	Failures
0	10.00000	4.487469	200	200	4.487473	0.000000	0
100	4.67624	2.584268	200	200	2.584496	0.000000	0
200	2.18672	2.636067	200	200	2.669750	0.000021	0
300	1.02257	2.265963	200	200	2.603463	0.000262	0
400	0.47818	1.901320	200	200	3.089631	0.001413	0
500	0.22361	1.445835	200	200	4.017063	0.006943	0
600	0.10456	1.017067	200	200	5.483668	0.024523	0
700	0.04890	0.690710	200	200	7.568390	0.013740	0
800	0.02287	0.493879	200	200	11.140062	0.000369	0
900	0.01069	0.344695	200	200	16.157481	0.000003	0
1000	0.00500	0.356638	200	200	32.602336	0.000000	0

Table 4. Dynamic manifold distance monitoring data during the reverse sampling process of the diffusion model (intrinsic dimension $d^* = 12$).

K. More Experiment Results

K.1. Numerical Verification of Geometric Biases for Two-Point on MLP

We also verify the two-point example in the main text using a standard MLP in place of RFNN. The purpose of this experiment is to check that the geometric picture from Section 3 is not specific to the RFNN parametrization. As in Section 5.2, the dataset is $\mathcal{D} = \{(-3, 0), (3, 0)\} \subset \mathbb{R}^2$, for which the ridge is the horizontal axis and the normal and tangent directions are explicit.

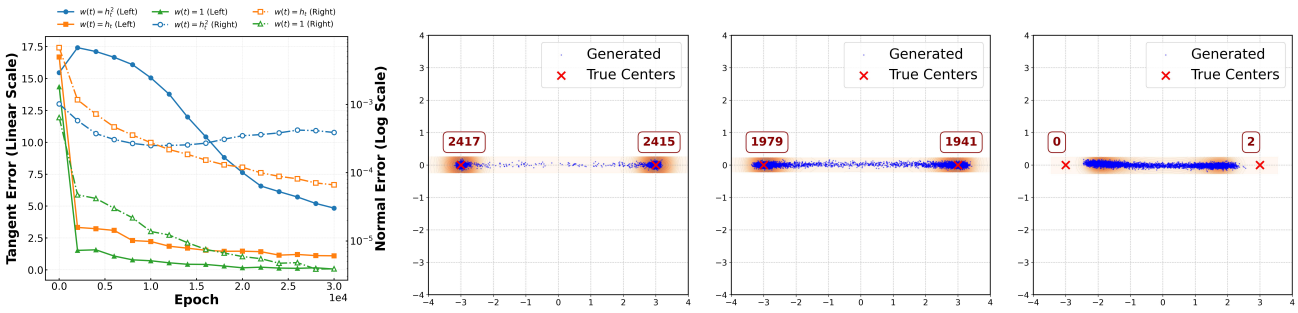
Model Architectures We employ a two-layer MLP for $m_A(x, t)$. The scalar t is first transformed via a sinusoidal embedding of dimension 32. This embedding is concatenated with the input $\mathbf{x} \in \mathbb{R}^2$, passed through a hidden layer of 128 units with ReLU activation, and finally projected to $\mathbb{R}^2$.

Training and Loss Computation We trained for 3×10^4 epochs with a learning rate of 1×10^{-4}. The gradient descent is performed using the full-batch SGD optimizer in PyTorch and loss function (5). We numerically evaluate the integral in the loss function (5) using the trapezoidal rule with 2,000 discretization points.

SDE and Sampling Details We model the diffusion process using a stochastic differential equation (SDE) with a drift coefficient of $f(x, t) = -x$ and a constant diffusion coefficient of $g(t) = \sqrt{2}$. For generation, we employ the Euler-Maruyama sampler with $N = 1000$ discrete time steps. The time schedule follows a geometric progression decaying from $T = 10$ down to $\delta = t_{min} = 10^{-3}$. Specifically, the time points are defined as $t_i = T \cdot (t_{min}/T)^{i/N}$ for $i = 0, \ldots, N$.

Figure 9 shows that the same directional geometric pattern observed in RFNN also appears in MLP. The normal error remains very small throughout training, and the generated samples stay tightly concentrated near the ridge $y = 0$. By contrast, the tangential error settles at a visibly larger floor, and the generated samples spread along the line segment between the two data points. Thus, even for MLP, the generated geometry is well explained by strong normal alignment together with persistent tangential spread.

The weighting schedule again mainly affects the tangential geometry rather than the normal geometry. In particular, $w(t) = 1$ yields the smallest tangential error and the strongest concentration near the two training points, while $w(t) = h_t$ and $w(t) = h_t^2$ produce progressively larger tangential floors and more pronounced edge-like interpolation. This is fully consistent with the interpretation in Remark 3.8 and shows that the directional geometric picture is not tied to the RFNN setting.



(a) Error Dynamics (b) $w(t) = 1$ (c) $w(t) = h_t$ (d) $w(t) = h_t^2$

Figure 9. **Error dynamics and generated samples of MLP.** (a) Evolution of tangential errors (solid lines, left axis, linear scale) and normal errors (dash-dot lines, right axis, log scale) during training. (b)–(d) Comparison of generated sample configurations under different weighting schedules. Boxed numbers indicate sample counts around the target modes (radius $= 0.5$). The background color represents the KDE plot.

K.2. RFNN on Different Sets

We consider additional 2D examples to illustrate that the ridge geometry continues to explain generation even when the geometry is no longer as simple as in the two-point case. The purpose of these experiments is not primarily quantitative verification, but to show that the proposed log-density ridge captures nontrivial low-dimensional generation patterns beyond a single straight line.

TWO-POINT CASE: (-3,0), (3,0)

We compare three distinct weight schedules: $w(t) = 1$ (Figure 10), $w(t) = h_t$ (Figure 11), and $w(t) = h_t^2$ (Figure 12). Across all three cases, the same qualitative three-stage evolution is visible, but the later tangential behavior changes

substantially with the weight schedule.

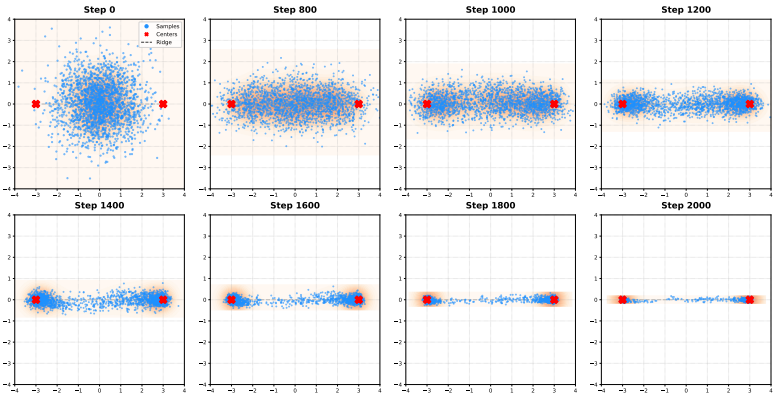


Figure 10. **Evolution of generated samples under the proposed sampling dynamics with weight schedule $w(t) = 1$.** The visualization displays snapshots of $N = 2000$ particles in a 2D plane during the sampling process. The background contours depict the kernel density estimation (KDE) of the particle distribution. Samples reach the ridge neighborhood at **Step 800**. **Steps 800–1800** illustrates Normal Alignment. **Steps 1800–2000** demonstrate Tangent Sliding.

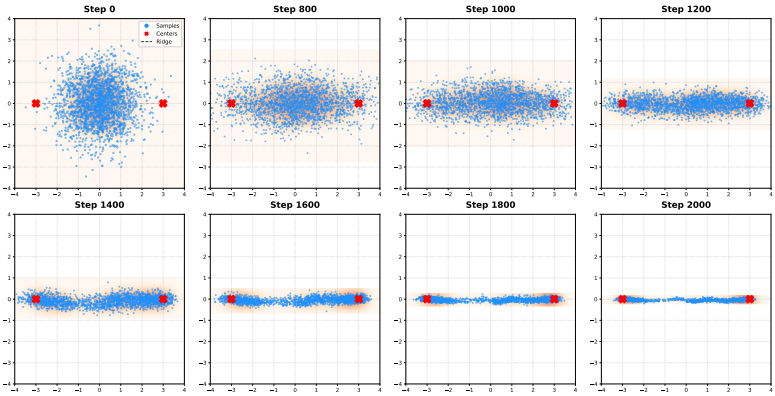


Figure 11. **Evolution of generated samples under the proposed sampling dynamics with weight schedule $w(t) = h_t$.** The visualization displays snapshots of $N = 2000$ particles in a 2D plane during the sampling process. The background contours depict the kernel density estimation (KDE) of the particle distribution. Samples reach the ridge neighborhood at **Step 800**. **Steps 800–1800** illustrates Normal Alignment. **Steps 1800–2000** demonstrate Tangent Sliding.

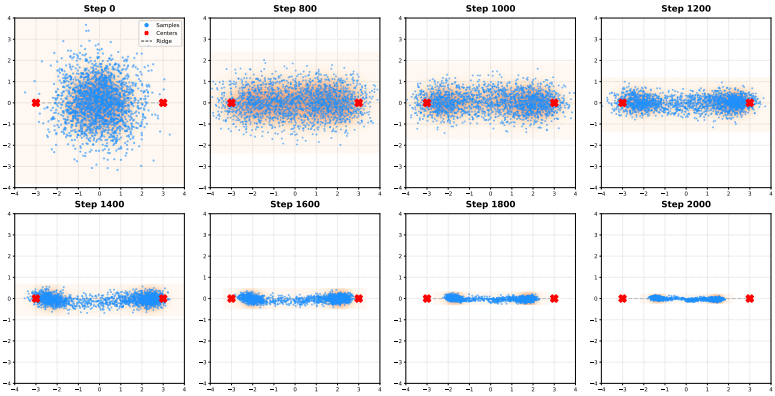


Figure 12. **Evolution of generated samples under the proposed sampling dynamics with weight schedule $w(t) = h_t^2$.** The visualization displays snapshots of $N = 2000$ particles in a 2D plane during the sampling process. The background contours depict the kernel density estimation (KDE) of the particle distribution. Samples reach the ridge neighborhood at **Step 1200**. **Steps 1200–1800** illustrates Normal Alignment. **Steps 1800–2000** demonstrate Tangent Sliding.

THREE-POINT CASE: (0,0), (5,0), (0,5)

This example shows that the theory is not limited to straight-line interpolation between two modes. Here the relevant ridge geometry is asymmetric and bent, so the figures illustrate how the same reach–align–slide mechanism persists even when the low-dimensional structure is no longer a single line segment. In particular, Figures 13, 14, and 15 show that for all three weight schedules $w(t) = 1$, $w(t) = h_t$, and $w(t) = h_t^2$, the late-stage motion follows the curved ridge induced by the three-point configuration rather than collapsing onto a straight interpolation path.

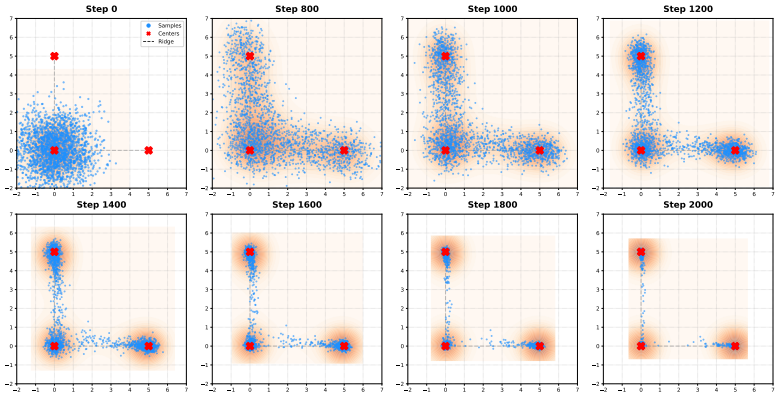


Figure 13. **Evolution of generated samples under the proposed sampling dynamics with weight schedule $w(t) = 1$.** The visualization displays snapshots of $N = 2000$ particles in a 2D plane during the sampling process. The background contours depict the kernel density estimation (KDE) of the particle distribution. Samples reach the ridge neighborhood at **Step 800**. **Steps 800–1400** illustrates Normal Alignment. **Steps 1400–2000** demonstrate Tangent Sliding.

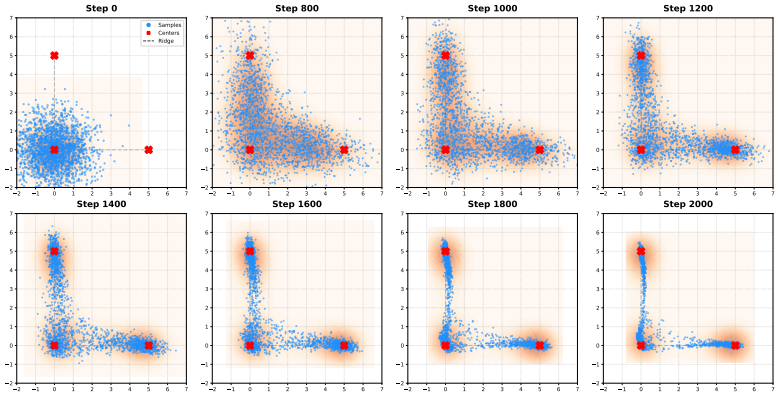


Figure 14. **Evolution of generated samples under the proposed sampling dynamics with weight schedule $w(t) = h_t$.** The visualization displays snapshots of $N = 2000$ particles in a 2D plane during the sampling process. The background contours depict the kernel density estimation (KDE) of the particle distribution. Samples reach the ridge neighborhood at **Step 1000**. **Steps 1000–1600** illustrates Normal Alignment. **Steps 1600–2000** demonstrate Tangent Sliding.

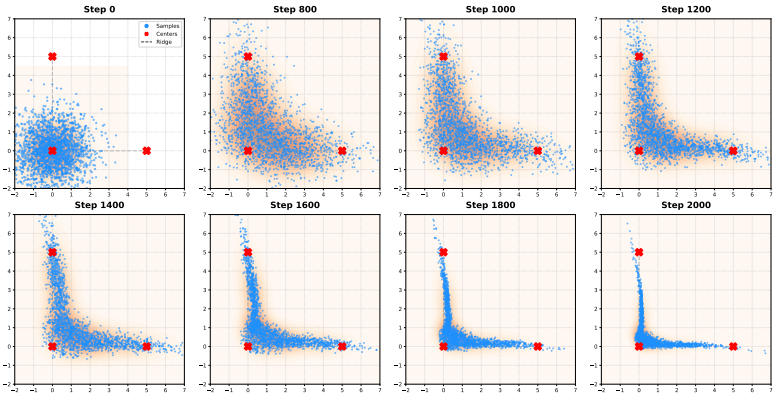


Figure 15. **Evolution of generated samples under the proposed sampling dynamics with weight schedule $w(t) = h_t^2$.** The visualization displays snapshots of $N = 2000$ particles in a 2D plane during the sampling process. The background contours depict the kernel density estimation (KDE) of the particle distribution. Samples reach the ridge neighborhood at **Step 1200**. **Steps 1200–1600** illustrates Normal Alignment. **Steps 1600–2000** demonstrate Tangent Sliding.

FOUR-POINT CASES

These examples probe more complicated ridge structures generated by four-point configurations. Figure 16 corresponds to the symmetric configuration $(\pm 2, \pm 2)$, where the ridge geometry is comparatively regular and the resulting sample evolution is correspondingly structured. Figure 17 shows an unsymmetric four-point configuration, where the ridge becomes more distorted and locally less regular. Figure 18 considers a qualitatively different arrangement in which three points form a triangle and the fourth lies inside that triangle, producing a more intricate interior geometry. Taken together, these examples show that the proposed ridge remains a useful reference object even when the local data structure becomes substantially more complicated than in the two- and three-point cases, and that the qualitative reach–align–slide behavior is still visible across these different configurations.

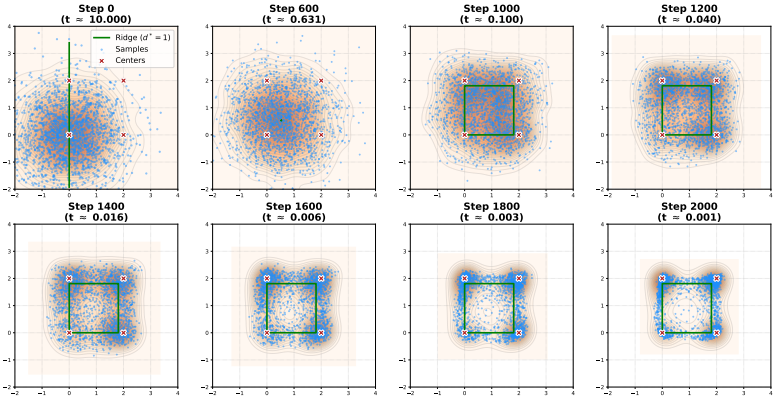


Figure 16. **Evolution of generated samples under the proposed sampling dynamics with weight schedule $w(t) = h_t$.** The visualization displays snapshots of $N = 2000$ particles in a 2D plane during the sampling process. The background contours depict the kernel density estimation (KDE) of the particle distribution. Samples reach the ridge neighborhood at **Step 1200**. **Steps 1200–1600** illustrates Normal Alignment. **Steps 1600–2000** demonstrate Tangent Sliding.

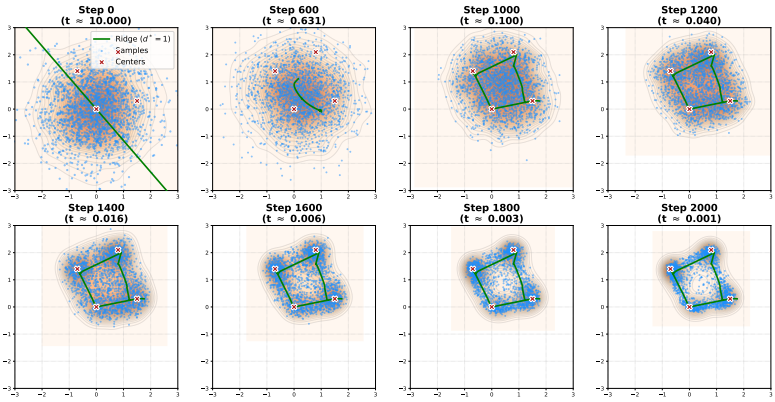


Figure 17. **Evolution of generated samples under the proposed sampling dynamics with weight schedule $w(t) = h_t$.** The visualization displays snapshots of $N = 2000$ particles in a 2D plane during the sampling process. The background contours depict the kernel density estimation (KDE) of the particle distribution. Samples reach the ridge neighborhood at **Step 1200**. **Steps 1200–1600** illustrates Normal Alignment. **Steps 1600–2000** demonstrate Tangent Sliding.

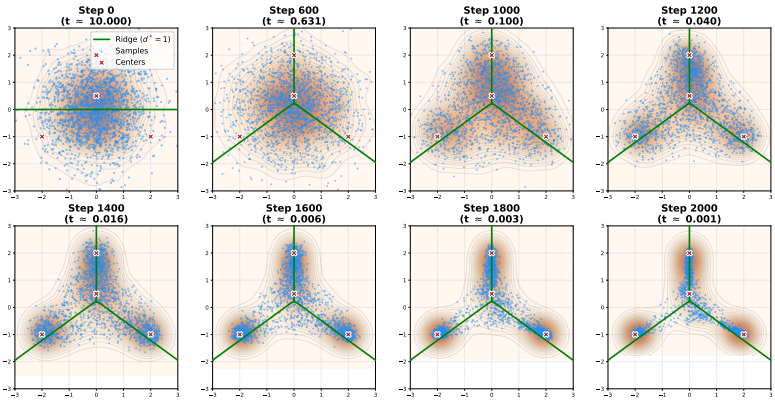


Figure 18. **Evolution of generated samples under the proposed sampling dynamics with weight schedule $w(t) = h_t$.** The visualization displays snapshots of $N = 2000$ particles in a 2D plane during the sampling process. The background contours depict the kernel density estimation (KDE) of the particle distribution. Samples reach the ridge neighborhood at **Step 1200**. **Steps 1200–1600** illustrates Normal Alignment. **Steps 1600–2000** demonstrate Tangent Sliding.

K.3. MNIST Trajectories

The figures in this subsection complement the quantitative MNIST results in the main text by showing the full visual evolution of generated samples. Figure 19 displays the overall sampling process, while Figure 20 focuses on the final 200 steps. The main point is that the large-scale semantic structure emerges relatively early, whereas the final stage of sampling produces only minor refinements. This is consistent with the main-text observation that normal alignment occupies most of inference, while late-stage tangential motion becomes very limited.

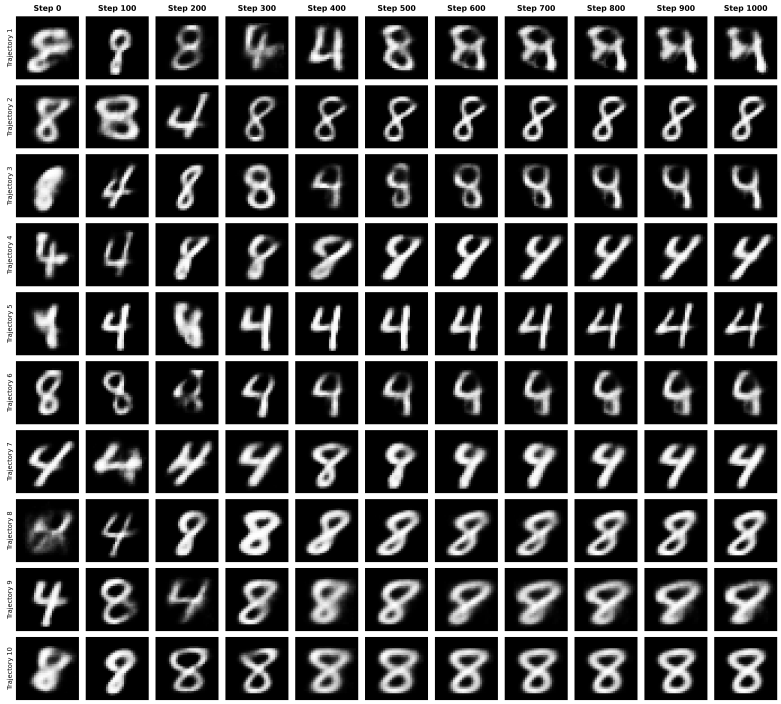


Figure 19. **Visualization of the full sampling evolution for 10 independent trajectories.** The process initiates from standard Gaussian noise at Step 0. Distinct geometric structures begin to emerge around Step 200 as the samples are pulled towards the ridge manifold. By Step 800, the digits (4 or 8) are clearly formed, showing that the semantic content has stabilized.

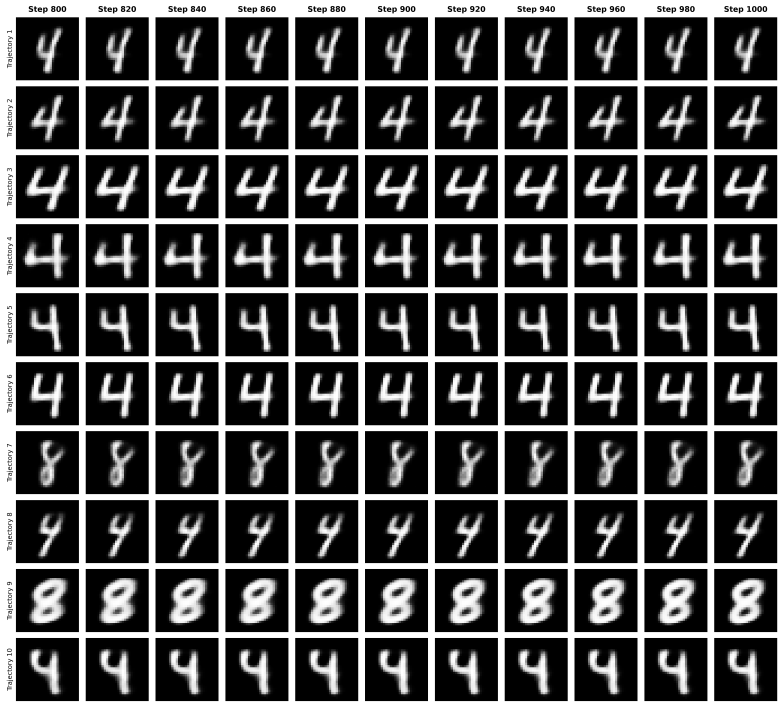


Figure 20. **Evolution during the final sampling phase (Steps 800–1000).** In this regime, the visual changes are negligible, restricted to minor high-frequency refinements. This visual stability corroborates our quantitative finding that the tangent direction error plateaus, indicating that the generated samples remain stationary on the manifold rather than drifting towards specific training data points.