
Supplementary material for “When and how can inexact generative models still sample from the data manifold?”

Anonymous Author(s)

Affiliation

Address

email

1 A Background on the random dynamical systems view of diffusion models

2 In the main text, we define the generating process of any generative modeling algorithm as a random
3 dynamical system. In particular, dynamical generative models like normalizing flow, rectified flow,
4 conditional flow matching-variants, stochastic interpolants [19, 15, 24, 16] etc can be viewed as
5 nonautonomous (forced in a time-dependent manner) and deterministic dynamical system. On the
6 other hand, diffusion models or score-based generative models [5, 26, 4, 17, 22] have stochastic
7 generating processes (reverse or denoising process). However, in the main text, we argued that, for the
8 purpose of analyzing the probability flow, we may ignore the noise in the reverse process, and thus,
9 also consider SGMs to be nonautonomous but deterministic systems. Here, we present a more general
10 dynamical systems definition applicable to both deterministic and stochastic nonautonomous systems.
11 These, so-called random dynamical systems have been classically studied as part of dynamical
12 systems theory, while undergoing parallel development in the probability and stochastic analysis
13 communities (see e.g., [3] and [12] for textbook expositions of random dynamical systems from the
14 dynamical systems/ergodic theory and probabilistic/stochastic analysis perspectives respectively).

15 The unifying random dynamical systems framework to represent generative models is as follows.
16 Consider an instance of a time-discretized Wiener path, $\Xi = \{\xi_0, \dots, \xi_{T-1}\}$, where ξ_i are indepen-
17 dent standard normal variables. These provide stochastic forcing to the dynamics, F_t^Ξ , at time t ,
18 which is now extended with a superscript Ξ to indicate a fixed noise path. For a fixed noise, Ξ , $F^{\tau, \Xi}$,
19 is defined as a composition (time τ -dynamics) as expected, $F^{\tau, \Xi} = F_{\tau-1}^\Xi \circ F^{\tau-1, \Xi}$, with $F^{0, \Xi} = \text{Id}$,
20 for all Ξ .

21 **Example: score-based diffusion [21, 22]** In score-based diffusion models, the generating process
22 is an Ito process of the form: $dX_t = f_t(X_t) dt + dW_t$, where f_t is a deterministic score term
23 that is represented as a neural network, and W_t is a Wiener path/Brownian noise. Given a time-
24 integration scheme for this stochastic process, we can define $F_{t, \Xi}$ as the stochastic flow over a
25 short time. For instance, using Euler-Maruyama time-integration with a fixed timestep, δt , we have
26 $X_{t+1} = X_t + f_t(X_t) \delta t + \sqrt{\delta t} \xi_t$. Then, $F_t^\Xi(x) := x + \delta t f_t(x) + \sqrt{\delta t} \xi_t$. In summary, we view a
27 stochastic generating process as a one-parameter family of random diffeomorphisms, F_t^Ξ , for (almost)
28 every time sampling, Ξ , of the underlying Brownian path. For the existence of this one-parameter
29 family, we refer to classical works on stochastic flows [12, 11]. With this RDS view, the stochastic
30 process (a continuous-state discrete-time Markov chain) has a time-dependent transition kernel that
31 can now be written in terms of F^Ξ as:

$$P_t(X_{t+1} \in A | X_t = x) = \mathbb{P}(\xi : F_t^\Xi(x) \in A).$$

32 Substituting for $F_t^\Xi(x) = x + \delta t f_t(x) + \sqrt{\delta t} \xi_t$, and using the fact that ξ_t has a normal distribution,
33 we get, $P_t(A|x) = \int_A e^{-\|y-x-\delta t f_t(x)\|^2/(2\delta t)} dy$ for this example process. The usual equation for
34 the evolution of the probability measures, say μ_t , is the Kolmogorov forward equation, which is given

35 by,

$$\mu_{t+1}(A) = \int P_t(A|x) d\mu_t(x) = \int \mathbb{P}(\xi_t : F_t^\Xi(x) \in A) d\mu_t(x). \quad (1)$$

36 On the right hand side of the above equation, notice the transition kernels written in terms of the
 37 RDS. Moreover, beyond μ_t , we can also define a sequence of *sample-path measures*, μ_t^Ξ , which are
 38 obtained for fixed Brownian paths via pushforwards or the *Frobenius-Perron* operator,

$$\mu_{t+1}^\Xi = F_{t+1}^\Xi \mu_t^\Xi := \mu_t^\Xi \circ F_t^{\Xi,-1}. \quad (2)$$

39 Since we start the process with $X_0 \sim \mu_0$, we take $\mu_0^\Xi = \mu_0$ for all paths Ξ . We note that since μ_0
 40 typically has a density (with respect to Lebesgue \mathbb{R}^d), say, ρ_0 , μ_t as well as the sample-path measures,
 41 μ_t^Ξ also have densities up to a finite time, even when F_t^Ξ is a non-volume preserving diffeomorphism.
 42 We denote these densities as ρ_t and ρ_t^Ξ respectively corresponding to μ_t and μ_t^Ξ . With the density ρ_t^Ξ
 43 defined, we can use the change-of-variables formula in (2) to obtain,

$$\rho_{t+1}^\Xi = \mathcal{L}_t^\Xi \rho_t^\Xi := \frac{\rho_t^\Xi \circ F_t^{\Xi,-1}}{|\det \nabla F_t^\Xi| \circ F_t^{\Xi,-1}}, \quad (3)$$

44 where we define \mathcal{L}_t^Ξ to be a time-dependent linear operator that transforms densities through a
 45 deterministic system. Combining with the Kolmogorov forward equation in (1), we also have,

$$\rho_{t+1}(y) = \mathbb{E}_\Xi \mathcal{L}_t^\Xi \rho_t^\Xi, \quad (4)$$

46 provided $\rho_0 = \rho_0^\Xi$, where we have used \mathbb{E}_Ξ to denote expectation with respect to the independent
 47 standard Gaussian RVs, $\Xi = [\xi_0, \dots, \xi_{\tau-1}]$.

48 For a fixed noise Ξ , the deterministic dynamics $F^{\tau,\Xi}$ is a coupling between ρ_0 and a noise path-
 49 dependent density ρ_τ^Ξ , i.e., $\mathcal{L}^{\tau,\Xi} \rho_0 = \rho_\tau^\Xi$. Here, the operator $\mathcal{L}^{\tau,\Xi}$ is called the *Frobenius-Perron* or
 50 transfer operator, which describes the evolution of probability densities through the map, $F^{\tau,\Xi}$. The
 51 Frobenius-Perron operator $\mathcal{L}^{\tau,\Xi}$ is also defined as a composition of per-iteration operators, which
 52 we denote by \mathcal{L}_t^Ξ , so that $\mathcal{L}_t^\Xi \rho_t^\Xi = \rho_{t+1}^\Xi$. Specifically, we define $\mathcal{L}_t^\Xi \rho = (\rho \Delta \text{vol}_t) \circ F_t^{\Xi,-1}$, where
 53 $\Delta \text{vol}_t(x) = |\det(dF_t^\Xi)|^{-1}$ indicates the change of differential volume under the application of the
 54 map F_t^Ξ . Note that, since the noise Ξ is independent of the state, Δvol_t is not a function of Ξ . In
 55 the standard stochastic analysis literature, we generally refer to $\mathbb{E}_\Xi \mathcal{L}_t^\Xi$ as the *Kolmogorov forward*
 56 operator, which is described in (1) when μ_t are absolutely continuous with respect to Lebesgue.
 57 By definition, $\mathbb{E}_\Xi \mathcal{L}_t^\Xi \rho_t = \rho_{t+1}$. Classically, we may write, $\rho_{t+1}(x) = \int_M \kappa_t(x, y) \rho_t(y) dy$,
 58 where $\kappa_t(x, y)$ is the conditional density of the transition kernel, $P_t(y, dx)$, which represents the
 59 conditional density of the state at time $t+1$ being x conditioned on the state at time t being y . This
 60 assumes the kernel is absolutely continuous in both arguments, which is typical for diffusion-based
 61 models (e.g., the transition probability measure in (1) is absolutely continuous). When the target
 62 measure, $\mu_\tau = p_{\text{data}}$ is singular, making ρ_τ undefined, the probability density $\rho_{\tau-\Delta}$ for a small Δ
 63 approximates a notion of density associated with the target. For simplicity, ρ_τ in this case should be
 64 interpreted as $\rho_{\tau-\Delta}$, which is a convolution of the target measure, p_{data} , with a Gaussian of variance
 65 Δ .

66 A.1 Diffusion models

67 Our paper treats the reverse process of a diffusion model as a random dynamical system. While we
 68 presented this view in the main text and the previous section, here we review the more standard view
 69 through SDEs. Diffusion models generate samples from an unknown *target* probability distribution
 70 $\pi \in \mathcal{P}(\mathbb{R}^D)$ from which we only have access to samples. The general setup [22] is to consider a
 71 diffusion process, which will be referred to as the *forward process*, that transforms the target into
 72 a distribution that is easy to sample from. Typically, the forward process is chosen from a class of
 73 Ornstein-Uhlenbeck processes

$$dX_t = -\beta_t X_t dt + \sqrt{2\beta_t} dB_t, \quad X_0 \sim \pi. \quad (5)$$

74 It is assumed that β_t is positive and integrable such that the integral $\int_0^t \beta_s ds \rightarrow \infty$ as $t \rightarrow \infty$. It
 75 follows that (5) is a time-rescaling of the standard Ornstein-Uhlenbeck process, through the time

change of variables $\tau = \int_0^t \beta_s ds$ and the marginals ρ_t converge geometrically to the standard multivariate normal distribution $N(0, I_D) \in \mathcal{P}(\mathbb{R}^D)$. Since (5) has linear drift, it follows that the solutions can be solved analytically, yielding the formula for the marginals in terms of the target $\rho_t(x) = \mathbb{E}_{X_0 \sim \pi}[\rho_t(x|X_0)]$ with the conditional density given by the kernel

$$\rho_t(\cdot|x_0) = N\left(\exp\left(-\int_0^t \beta_s ds\right)x_0; \left(1 - \exp\left(-2\int_0^t \beta_s ds\right)\right)I_D\right). \quad (6)$$

We note here that the above kernel is smooth in the space variable, implying the C^∞ smoothness for the marginals for all $t > 0$.

The forward process is ergodic, with the marginals converging to the standard normal at rate $\exp\left(-\int_0^t \beta_s ds\right)$. After a finite large time T samples are assumed to be approximately normal. Once T is chosen, define the time-reversed process $Y_t := X_{T-t}$, $t \in [0, T]$. It is known ([7], [2]) that Y_t is a Markov process and that it is a weak solution to the following stochastic integral

$$dY_t = \beta_{T-t}(Y_t + 2\nabla \log \rho_{T-t}(Y_t)) dt + \sqrt{2\beta_{T-t}} dB_t, \quad Y_0 \sim \rho_T. \quad (7)$$

From the smoothness of the marginals ρ_t — and hence smoothness of the drift term $y + 2\nabla \log \rho_t(y)$ — it follows that (7) admits a strong solution for times $t < T$. It follows from weak uniqueness of the backward process [18] that the law of Y_{T-t} coincides with that of X_t . Hence, generating trajectories from the reverse process provides a way of sampling from the target distribution as $t \rightarrow T$.

The backward process is only defined for times $t < T$. In order to extend the backward process to the full time interval $t \in [0, T]$, one needs the assumption on the initial density that $\nabla \log \rho_0 = \nabla \log \pi$ exists in a weak L^2 sense [7]. However, in practice this is almost never satisfied as the target density is typically singular. This implies a singularity in the score s_t that grows as $\frac{1}{\int_0^t \beta_s ds}$ as $t \rightarrow 0$. To overcome this issue, the backward process is typically only sampled up to time $t = T - \Delta$, which is responsible for the characteristic noise typically present in image models.

B Response of the predicted density to learning errors

In the main paper, we argued that we may ignore the noise term ξ for a fixed path in probability space, and simply consider the deterministic nonautonomous system. Here we show how to extend the perturbation response result in section 3 to random dynamical systems. Using the framework presented in section A, we may go through the same derivation as in section 3 pathwise, by replacing \mathcal{L}_t with \mathcal{L}_t^Ξ . Again, the density ρ_τ^Ξ , is close to the target (on averaging with respect to the noise paths, Ξ), but not exactly equal. In case the target density with respect to Lebesgue does not exist, we can perform integration by parts and treat ρ_τ^Ξ as a genuine density due to the convolution of the target measure with Gaussians that describes the ρ_τ^Ξ in the discrete time algorithm (DDPM).

As before, we consider f to be constant functions on the data manifold that are differentially extended to \mathbb{R}^d . The pathwise responses derived in this way contain pathwise score functions, s^Ξ which are not the same as the score functions, s . While $\mathbb{E}\rho^\Xi = \rho$, we do not get the score by taking expectations of the pathwise scores. In order to compute s^Ξ however, we may recursively apply the log gradient of the change of variables through the map, F^Ξ , i.e., \mathcal{L}_t^Ξ . The above pathwise statistical response, if uniformly bounded over the Wiener paths, due to dominated convergence, allows us to exchange limits, and thus, the overall statistical response can still be computed pathwise via,

$$\langle f, \partial_\epsilon|_{\epsilon=0} \mathbb{E}_\Xi \mathcal{L}_\epsilon^{T,\Xi} \rho_0 \rangle = \langle f, \mathbb{E}_\Xi \partial_\epsilon \mathcal{L}_\epsilon^{T,\Xi} |_{\epsilon=0} \rho_0 \rangle. \quad (8)$$

C Tangent dynamics: evolution of infinitesimal perturbations

The primary objective of this work is to study the effect of learning errors on the dynamics. For stochastic generative processes, we can extend the linear perturbation analysis in the main text to each noise realization of an RDS. As before, to model the effect of score learning errors, we consider evolving $F_{t,\Xi}$ with perturbed scores of the form, $s_t + \epsilon \chi_t$, where χ_t is a time-dependent vector field that indicates the direction of the error in the score. The perturbed dynamics, for a fixed noise path, is represented as, $F_\epsilon^{t,\Xi} = F_{t-1,\epsilon}^\Xi \circ \dots \circ F_0^\Xi$, and correspondingly, the perturbed densities, by $F_\epsilon^{t,\Xi} \rho_0 = \rho_{t,\epsilon}^\Xi$, leading to the perturbed predicted density, $\rho_{\tau,\epsilon}$, when we take an expectation over

noise realizations Ξ . We can set $\zeta_t := \partial_\epsilon F_\epsilon^{t,\Xi}$ to represent a time-dependent vector field that gives the perturbation in the state (sample) at time t due to the learning error field. Taking $\epsilon \rightarrow 0$, we can obtain the following recursive relationship for ζ_t :

$$\zeta_{t+1} \circ F_t^\Xi = dF_t^\Xi \zeta_t + \chi_t \circ F_t^\Xi, \quad (9)$$

simply by applying chain rule. Unrolling this recursion, and since $\zeta_0^\Xi = \partial_\epsilon F_\epsilon^{0,\Xi} = \partial_\epsilon \text{Id} = 0$ identically as a vector field, we obtain,

$$\zeta_{t+1}^\Xi \circ F_t = \sum_{n=0}^t dF_t dF_{t-1} \circ F_{t-1}^{-1} \cdots dF_{n+1} \circ F_{n+1}^{-1} \circ \cdots \circ F_{t-1}^{-1} \chi_n. \quad (10)$$

A vector field can be evaluated at a specific point, say $x \in \mathbb{R}^D$, to give a *tangent vector*, that indicates the direction of infinitesimal change at x . An interpretation of this infinitesimal change when viewed through differentiable scalar fields is the following. If $g : \mathbb{R}^D \rightarrow \mathbb{R}$ is a scalar function on the domain, then, at x , a vector field represents one among the possible directions of an infinitesimal change in g . In other words, tangent vector fields can be thought of as (linear) operators which when acting on differentiable functions produce their directional derivatives at each point. As an example, $\zeta_t(x) \in \mathbb{R}^D$ is a tangent vector that can be used to produce the directional derivative of any g , as $dg(x) \cdot \zeta_t(x) := \lim_{\epsilon \rightarrow 0} (g(x + \epsilon \zeta_t(x)) - g(x)) / \epsilon$. In this sense, there is a natural interpretation for the sequence of vector fields defined in (9). Let us fix an orbit/sample path, $\{x_t = F_t^\Xi(x_{t-1})\}$. The tangent vectors $\zeta_t(x_t) \in \mathbb{R}^D$ can be applied to a scalar function g to obtain the overall infinitesimal change in g along the sample path due to infinitesimal learning errors. More precisely,

$$\partial_\epsilon (g \circ F^{t,\Xi})(x_0) = dg(x_t) \cdot \zeta_t(x_t). \quad (11)$$

Rewriting (10) to make $\zeta_t(x_t)$ explicit along a fixed sample path,

$$\zeta_t(x_t) = \sum_{n=0}^{t-1} dF_{t-1}(x_{t-1}) \cdots dF_{n+1}(x_{n+1}) \chi_n(x_{n+1}). \quad (12)$$

Each term in the above sum consists of multiplication by a sequence of matrices. Let us define $A_t := dF_t(x_t) \in \mathbb{R}^{D \times D}$ and the product $A_{n,t} := A_t A_{t-1} \cdots A_n$, for $0 \leq n \leq t-1$, for the sake of shorter notation. That is, the perturbation vector $\zeta_t(x_t)$ can now be written as

$$\zeta_{t+1}(x_{t+1}) := \sum_{n=0}^t A_{n+1,t} \chi_n(x_{n+1}). \quad (13)$$

To analyze the effect of infinitesimal errors on infinitely long sample paths, we can let $n \rightarrow -\infty$ in the above equation. In this case, the asymptotic behavior of the product of random matrices comes into play. Oseledets theory (see e.g., [3]) is a collection of classical results on random matrix products as applied to cocycles defined on dynamical systems. Essentially, assuming that $\max\{0, \log \|A_t\|\}$ is summable for almost all paths, one can define Lyapunov exponents (for each Ξ) to be the logarithms of the set of eigenvalues of the matrix, $\lim_{n \rightarrow -\infty} (A_{n,t}^\top A_{n,t})^{1/2(t-n)}$. Corresponding to the Lyapunov exponents (LE), there is a decomposition of the tangent space at each t in the characteristic directions called Oseledets subspaces, i.e., directions in which the perturbation norms grow at an exponential rate corresponding to a given LE. Thus, to analyze the growth/decay of the norms in the time-dependent linear dynamical system (9), these characteristic directions form a natural basis. Here, since our dynamical system is defined only over a finite time interval, we consider a computational proxy for the Oseledets spaces, which are described in the main text (section 4). In the remainder of this section, we let $n \rightarrow -\infty$ and review Oseledets theorem.

Ignoring the control or forcing (inhomogeneous) term in 9, to isolate the time-asymptotic growth/decay on an exponential scale, we can consider the following homogeneous tangent equation,

$$\omega_{t+1} = A_t \omega_t. \quad (14)$$

If we are only interested in growth/decay on an exponential (in t) scale, finite sums for n close to t in (13) are not significant. Moreover, the vectors $\chi_n(x_{n+1})$ are path-dependent and perturbation-dependent, and they are not fundamental directions characteristic of the dynamics. Thus, by considering a decomposition (as in section 4) of $\chi_t(x_t)$ along Oseledets spaces, we can provide a pessimistic

analysis, since a random vector $\chi_t(x_t)$ will, with probability 1, have a non-zero component in the leading Oseledets space at x_t .

The homogeneous tangent equation (14) gives the evolution of infinitesimal perturbations in the initial conditions, i.e., $\omega_t := dF^{t,\Xi} \omega_0$. This equation gives the most general evolution of infinitesimal perturbations along a generic sample path $\{x_t\}$. When x_t is invariant, i.e., a fixed point, A_t is also invariant, and this reduces to linear stability analysis. When x_t is a periodic orbit, the matrices A_t are classically studied with Floquet theory and corresponding exponents. In more generality, the random matrix product $A_{n,t}(x_n) : T_{x_n} \mathbb{R}^D \rightarrow T_{x_t} \mathbb{R}^D$, known as the *tangent propagator* [13], is studied as $n \rightarrow -\infty$ under Oseledets multiplicative ergodic theorem.

When the dynamics F_t is invertible, we consider the limit

$$W^-(t) = \lim_{n \rightarrow -\infty} (A_{n,t}^{-\top} A_{n,t}^{-1})^{1/(2(t-n))}.$$

The eigenvectors $\phi_i(t)$ of $W^-(t)$ are called the *backward Lyapunov vectors (BLVs)*, and the negative log of the singular values $\lambda_i = -\log \sigma_i$ are called the Lyapunov exponents. Conventionally, the LEs are still deterministic and are defined by taking expectations with respect to the noise paths Ξ . The vectors $\phi_i(t)$ form a basis for the tangent space at x_t are defined for \mathbb{P} -a.e. (for almost every noise realization). For an exposition on the ergodic theory for RDS, see [10]. When the distribution \mathbb{P} does not depend on time, the Backward Lyapunov vectors can also be defined in a deterministic manner $\mathbb{P} - a.e.$

D Robustness of the support upon alignment

Proposition 4.1 shows that with high probability an aligned and convergent generative model can be used to learn the support of the data distribution accurately. First, by convergence, we mean that the generating process enjoys a theoretical convergence result in Wasserstein metric. For instance, we can consider a convergence result from [14] for denoising diffusion probabilistic model (DDPM), a time-discrete diffusion model. For any general target with compact support, as we have assumed, suppose the score is learned with a L^2 error $\mathcal{O}(\epsilon)$, uniformly over time $t \leq \tau$. Then, [14] show that the Wasserstein-2 distance between the predicted density, $\rho_{\tau,\epsilon}$ and the target p_{data} is $\mathcal{O}(\epsilon^{1/18})$. Note that, by definition of Wasserstein-2 distance, if T is the optimal transport map between p_{data} and $\rho_{\tau,\epsilon}$, then, $E_{x \sim p_{\text{data}}} \|T(x) - x\|^2 \leq C\epsilon'$, where $T(x) \sim \rho_{\tau,\epsilon}$. Now since $\|T(x) - x\|$ is a random variable with a small mean and variance, we can get an ϵ_0 (applying Chebyshev's inequality e.g.) in the statement of Proposition 4.1 given any $\delta > 0$, such that with probability (over n independent draws from p_{data}) $\geq 1 - \delta/2$, we have that $\|T(x_i) - x_i\| \leq \epsilon_0$, for all $i \leq n$.

Next we examine the alignment property. In Proposition 4.1, we assume alignment with high probability. That is, with probability $\geq 1 - \delta/2$ over independent draws from p_{data} , alignment holds, i.e., at the generated samples, $T(x_i)$, the most sensitive Lyapunov subspace E^d is tangent to the support of p_{data} . In other words, the generated samples $T(x_i) = x_i + \epsilon h_i$, where h_i is along $T\partial M$. Now consider a one-classifier trained to predict 1 if a data point is on the support and -1 otherwise. A kernel-based classifier is always realizable for a discrete data distribution [20]. It is a one-class classifier because for all the data points x_i , the output label is 1 and we do not have negative samples.

A key observation is that the confidence margin of a (one-class) hyperplane classifier trained using x_i is the same as that trained using $T(x_i)$. Therefore, we can apply a known generalization result, and going further, even data-dependent upper and lower bounds for classification using the true data distribution to now the predicted distribution, provided the prediction is aligned (margin does not change). This is the essence of the proof. In summary, we pose learning the support as estimating a one-class classifier. Then, we use the fact that the margin does not change when we move data points along the separating hyperplane.

E Alignment proofs

In the proof of Theorem 4.3, we make assumptions about the dynamics of the vector field v_t , whose time-discretized flow is our dynamics, F^t . Mainly, toward the end, when $t > t^*$, we assume specific anisotropic behavior of the vector field. It is helpful to think of the anisotropy by considering local coordinates that align the first d coordinates with the most sensitive subspaces, E_t^d . In other words,

consider local coordinates, $\Phi_t : \mathbb{R}^D \rightarrow \mathbb{R}^D$ around x_t , such that, $\Phi_t(0) = x_t$ and $d\Phi_t(0)$ maps the first d standard basis vectors to E_t^d .

Recall assumptions (i)-(iii) in the statement of Theorem 4.3. Consider the Jacobian matrix at time t , $dF_t(x)$ in block form, $dF_t(x) = \begin{bmatrix} \text{Id} + \delta t \nabla_{t,d} v_{t,d}(x) & \delta t \nabla_{t,d\perp} v_{t,d}(x) \\ \delta t \nabla_{t,d} v_{t,d\perp}(x) & \text{Id} + \delta t \nabla_{t,d\perp} v_{t,d\perp}(x) \end{bmatrix}$, and the second derivative $d^2 F_t$ can be written as two block tensors: $\begin{bmatrix} \delta t \nabla_{t,dd}^2 v_{t,d}(x) & \delta t \nabla_{t,dd\perp}^2 v_{t,d}(x) \\ \delta t \nabla_{t,dd}^2 v_{t,d\perp}(x) & \delta t \nabla_{t,dd\perp}^2 v_{t,d\perp}(x) \end{bmatrix}$ and $\begin{bmatrix} \delta t \nabla_{t,dd\perp}^2 v_{t,d}(x) & \delta t \nabla_{t,dd\perp}^2 v_{t,d\perp}(x) \\ \delta t \nabla_{t,dd\perp}^2 v_{t,d\perp}(x) & \delta t \nabla_{t,dd\perp}^2 v_{t,d\perp}(x) \end{bmatrix}$. To obtain an estimate of $w_t := \text{tr}((dF_t)^{-1} d^2 F_t)$, we first observe that using assumptions (ii)-(iii), the Schur complement of the first $d \times d$ block of dF_t reduces to $\text{Id} + \delta t \nabla_{t,d\perp} v_{t,d\perp}$. Using this Schur complement and assumption iii, we obtain that the first block of w_t , which is $w_t E_t^d$ is given by $\delta t \text{tr}((\text{Id} + \delta t \nabla_{t,d} v_{t,d})^{-1} \nabla_{t,dd}^2 v_{t,d})$. Then, using assumption ii, we obtain the estimate in the main text.

F Regularity of alignment

Lemma 4.4 shows a notion of regularity of the alignment property. We show this by using the Arzela-Ascoli theorem on the space of functions E_ϵ^d , for some ϵ perturbation of the dynamics. Applying Arzela-Ascoli gives the existence of a converging subsequence on this space. This subsequence consists of most sensitive subspaces of perturbed systems, which from convergence, will also be closely aligned with the data manifold if the original dynamics is aligned. To apply Arzela-Ascoli, one sufficient condition is to assume $F_{t,\epsilon} \in C^{1+\alpha}$ since we then obtain that E_ϵ^d is Holder continuous. This is because E_ϵ^d is by construction an orthonormal basis for the column space of dF_ϵ^t , which is C^α . For the Holder continuity of E_ϵ^d , we also need the eigenvalues of dF_ϵ^t to be nondegenerate. With uniform Holder constants and exponents, since M is compact, we get the needed equicontinuity.

G Additional numerical experiments

Our numerical results using score-based diffusions indicate robustness of support in all cases; further, they also show alignment, qualitatively validating the dynamical mechanism for robustness that we show in the main text. We report on the numerical methods, implementation details and our experiments in this section. The supplementary material also contains the code needed to reproduce the figures in the main text.

G.1 Sampling via reverse diffusion

In the case of score-based diffusions, our dynamics F^τ refers to the Euler-Maruyama discretization of the reverse diffusion (7). There are various noise schedules β_t used in practice. In terms of the continuous time SDE (5), choosing β_t is tantamount to reparameterizing the time variable in the standard Ornstein-Uhlenbeck process via $\tau = \int_0^t \beta_s ds$. From a mathematical perspective, the density evolutions are therefore the same. Practically, however, the process has to be discretized and some noise schedules are more robust against time-discretization errors [8]. For the purpose of this study, we therefore fix the noise schedule to be the cosine noise schedule from [17] that was shown empirically to yield good FID and NLL scores. We observe that our experimental results on alignment and robustness do not change when using different noise schedules. The cosine noise schedule from [17] translates to the formula for β_t given by

$$\beta_t = \frac{\pi}{(1+\delta)} \cdot \frac{\sin\left(\frac{\pi}{2} \cdot \frac{t+\delta}{1+\delta}\right)}{\cos\left(\frac{\pi}{2} \cdot \frac{t+\delta}{1+\delta}\right)}.$$

This comes from the formula for $\bar{\alpha}_t = f(t)/f(0)$, $f(t) = \cos\left(\frac{t+\delta}{1+\delta} \cdot \frac{\pi}{2}\right)^2$ given in [17] and noting that $\bar{\alpha}_t = \exp\left(-\int_0^t \beta_s ds\right)$.

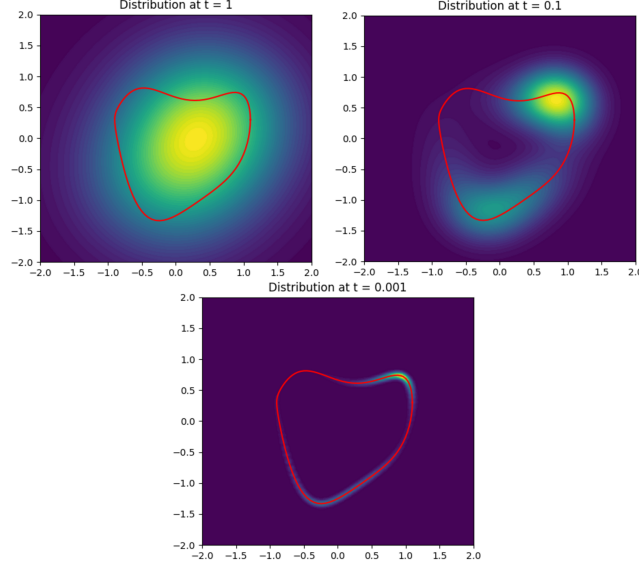


Figure 1: Score-based diffusion with numerical estimates of the score. Top row: starting density, ρ_0 and the density at time 0.9. Bottom row: predicted target, $\rho_{1-\Delta}$. In each figure, the red curve represents the analytical data manifold.

Once a suitable approximation to the score is acquired, the backward equation (7) is discretized to yield the random dynamical system

$$Y_{n+1} = F_n(Y_n, \xi_n) := Y_n + \beta_{T-t_n} (Y_n + 2s_{T-t_n}(Y_n)) \delta t + \xi_n \sqrt{2\beta_{T-t_n} \delta t}, \quad Y_0 \sim N(0, 1).$$

We also study solutions the *perturbed* system

$$Y_{n+1} = F_n(Y_n, \xi_n) := Y_n + \beta_{T-t_n} (Y_n + 2s_{T-t_n}(Y_n) + \epsilon \chi_{T-t_n}(Y_n)) \delta t + \xi_n \sqrt{2\beta_{T-t_n} \delta t}.$$

The perturbation vector χ_n specifying the error between the original dynamical system $F_n(\cdot, \xi_n) = F_n(\cdot, \xi_n; 0)$ and the perturbed dynamical system $F_n(\cdot, \xi_n; \epsilon)$, and ϵ measures the strength of the perturbation (see A). The timesteps t_n is chosen equispaced with $0 < t_0 < \dots < t_n = T - \Delta = 1 - \Delta$, with Δ controlling the early stopping time to avoid singularities. This corresponds to solving the backward SDE (7) from $t = T - t_0$ backward to $t = \Delta$.

G.2 Two dimensional examples

We perform a number of experiments on two-dimensional domains with one or two-dimensional support of the target. We show our experiments with the 2 moons distribution in Figures 1(main paper), 6 and 5. We also visualize the Lyapunov vectors on a different example in Figures 1 and 2. Throughout, LVs and LEs are computed using the QR algorithm (a finite-time version of the Gram-Schmidt process from [6]) described in section 4.

In these planar experiments, we represent the manifold as a curve (or a collection of curves as in the half-moon example) $\Gamma = \{\Gamma(t) : t \in [0, 1]\} \subset \mathbb{R}^2$. The target measure is given by $dp_{\text{data}} = q d\gamma$ where $d\gamma$ is the arc-length measure for the curve $\Gamma(t)$, and q some smooth density. We can compute expectations against π via the parameterization as

$$\mathbb{E}[g(X)] = \int_{\Gamma} g(x) p(x) ds = \int_0^1 g(\Gamma(t)) p(\Gamma(t)) \Gamma'(t) dt.$$

The Ornstein-Uhlenbeck process 5 is a linear SDE with additive noise. The density ρ_t can therefore be solved analytically [18] via the integral equation

$$\rho_t(x) = \int_{\Gamma} \rho_t(x|x_0) q(x_0) ds.$$

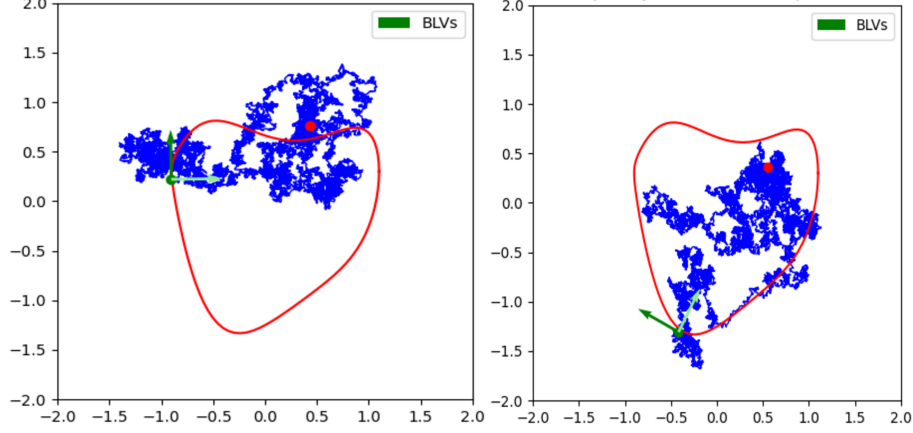


Figure 2: The finite-time Lyapunov vectors shown in green at two samples of the predicted distribution. The reverse sample path is shown in blue. The leading LV, shown in a darker green, shows alignment with the data manifold.

The kernel $\rho_t(x|x_0)$ is the Green’s function to the associated Fokker-Planck equation and is given by

$$\rho_t(x|x_0) = \frac{1}{Z_t} \exp \left(-\frac{|x - e^{-\frac{t}{2}} x_0|^2}{2(1 - e^{-t})} \right).$$

The score $s_t = \nabla \log \rho_t$ can also be solved for in terms of the one dimensional integral

$$s_t(x) = \frac{\int_{\Gamma} \nabla_x \rho_t(x|x_0) q(x_0) ds}{\int_{\Gamma} \rho_t(x|x_0) q(x_0) ds}.$$

Our paper focuses on the propagation of score errors through the dynamics. To validate our theoretical results on the robustness of the support in a stylized setting, and since the integrals involved are tractable in the low-dimensional setting, we estimate the score via quadrature rather than training a neural network. This is done to maintain explicit control of the errors involved in our motivating examples and experiments.

G.3 MNIST training details

Here we present additional details on the MNIST results from the main paper. We showed that MNIST generation with diffusion models tends to have robustness of the support. Further, we also observed that our proposed mechanism of alignment holds even in this higher dimensional setting. Specifically, we showed that the leading $\mathcal{O}(20)$ (approximately the intrinsic dimension of the support/data manifold) LVs span the tangent spaces to the data manifold. As empirical proof of this, we saw that moving along an LV of a higher index (indices are, by convention, in decreasing order of LEs) takes us off the data manifold. This is shown in more detail in Figure 3.

These images are produced by a DDPM where the score model is trained by minimizing the simplified conditioned score matching loss from [9]:

$$\mathcal{L}_{\text{simple}}(\theta) = \mathbb{E}_{t, x_0, \epsilon} [\|\epsilon - \epsilon_{\theta}(\sqrt{\alpha_t} x_0 + \sqrt{1 - \alpha_t} \epsilon, t)\|^2],$$

where $t \sim U(\sigma_{\min}, T)$, $x_0 \sim p_{\text{data}}$ and $\epsilon \sim \mathcal{N}(0, I)$. Once again we note that in the continuous setting we have $\alpha_t = \exp \left(-\int_0^t \beta_s ds \right)$. Training is done in batches of 64 for 30 epochs. The backward process consists of 4000 steps, generating a trajectory of (11) from time $T = 0.9$ down to $\Delta = T/4000$. We use an Adam optimizer with learning rate $2e-5$.

Once trained, the score approximation is given by $s_{\theta}(x, t) = \frac{1}{\sqrt{1 - \alpha_t}} \epsilon_{\theta}(x, t)$. The neural network ϵ_{θ} is a U-Net, that was implemented in PyTorch by [25]. The U-Net consists of two down-sampling stages, one mid-level stage, and two up-sampling stages, where the 28×28 image is down-sampled to an array of 7×7 images and up-sampled again. Each downsampling stage consists of two ResNet

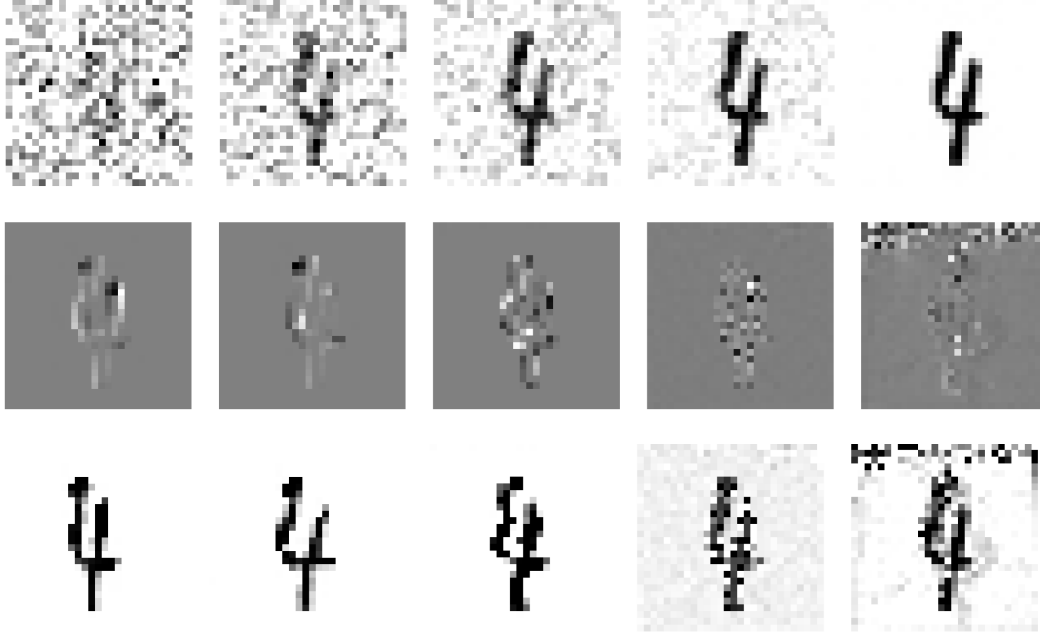


Figure 3: Top: Denoising diffusion trajectory sampled by approximating the score using a U-Net architecture trained on the MNIST digit dataset. Middle: The Lyapunov vectors of indices 1, 2, 20, 50 and 100 (from left to right) calculated along the sample trajectory. Notice that the principal Lyapunov vectors recover meaningful features of the sampled digit. This is in contrast to the lower Lyapunov vectors (higher indices indicate smaller LEs), which become progressively more noisy. Bottom: The sample image perturbed in the direction of the Lyapunov vectors in the same column. The Lyapunov vectors represent the principal directions in which errors in the sampling algorithm influence the sampled image. Notice that the principle Lyapunov vectors morph the shape of the sample without destroying image fidelity, whereas the lower Lyapunov vectors destroy image structure. This is consistent with our claim that errors propagate the image tangent to the data manifold.

layers with SiLU nonlinearity and an Attention layer. The mid-level consists of a ResNet layer followed by an Attention layer followed again by a ResNet layer, before up-sampling in a symmetric fashion.

G.4 CIFAR-10

Our perturbation experiments on the CIFAR-10 data distribution also confirm the robustness of the support property exhibited by score-based diffusion models. In Figure 4 (left), we show images sampled by a pretrained generative model from [23] Github. On the right hand side of Figure 4, we show images generated by the same model with a score perturbation of size 0.1 (L_∞ norm) added to F_t for each t . These samples look visually no different and produce similar likelihood scores, ≈ 3.7 bits/dim, compared to the predicted samples using the original pretrained score model even for perturbation size up to 1. As expected, this behavior of robustness of the support is reproduced with any stable time-integration scheme, e.g., using predictor-corrector or probability flow ODEs.

G.5 Conditional flow matching

So far, all our numerical experiments were carried out with diffusion models. Here we compare the robustness of the support across other conceptually different generative models. Specifically, we consider experiments with conditional flow matching variants [16, 15] and stochastic interpolants [1], and all our experiments are based on the implementation by the TorchCFM package Github. At their core, these dynamical generative models interpolate samples from a source density ρ_0 and samples from the target p_{data} . For instance, a variance-preserving interpolation is used in stochastic interpolants [1] and a straight line interpolation is proposed in rectified flow sampling [16]. In these



Figure 4: Left: images predicted by a pre-trained score-generative model by Song et al 2021 [Github link] on CIFAR-10 training images. Right: Predicted images by the model when a size 0.1 perturbation is added to the score vector field.

generative models, a stochastic path such as $X_t = (1 - t)X_0 + tX_1 + \sigma(t)\xi_t$, with $X_0 \sim \rho_0$, $X_1 \sim p_{\text{data}}$ is predetermined, while the probability flow path is computed. This is in contrast to SGMs, where the probability path is predetermined for the reverse process by choice of the forward process. The score approximation is performed for Brownian/OU paths in SGMs, while for other paths in flow matching. Thus, it is natural to ask if the learned dynamics for these different probability paths also possess the robustness property.

Less robust flow matching models. Following TorchCFM tutorials [24], we learn vector fields v_t with an MLP and 256 training samples per epoch from the 2 moons data distribution. The generated probability density is quite accurate for all of these models. In Figure 5 (top left), we show the generated density from Optimal Transport-Conditional Flow matching [24]. Next, we add a perturbation of size 0.5 and 1.0 in the L^∞ norm to the learned vector field v_t . The predicted densities for OT-CFM (first row) and stochastic interpolants (third row) seem to show the most robustness to the support, while for non-rectified flow matching the densities do not seem to exhibit robustness of the support, in comparison. Visually, all of these models seem to be less robust (c.f. Figure 1) than score-based diffusions. It is noteworthy that this is not due to the effect of the noise in the diffusion process, as the same robustness is visible even for deterministic time-integration (probability flow ODEs) using the scores. Thus, the robustness seems to be dynamical, with different dynamics on probability space and the loss function/formulation together dictating specific dynamics on sample space.

To understand the effect of the dynamics further, we compute the LVs and LEs as before using an iterative QR algorithm. Recall that the LEs are recovered as the time-average of the log diagonal elements of R_t (see section 4 of the main paper). We observe that some paths (i.e., with non-zero probability with respect to the source distribution) may have positive leading LEs, while SGMs were always observed to have stable LEs. We take the source density to be 8 Gaussians, but essentially similar results were obtained with a standard Gaussian source density.

In Figure 6, we show the leading LV (in blue) calculated for three different GMs in the top row. Also plotted is the score of the approximate 2 moons density (shown in red) in each case. The model OT-CFM seems to be most consistent with Theorem 4.3, showing most orthogonality with the score, or alignment, among the flow-matching models, but much less compared with diffusion models. To quantify the alignment, we plot the histogram of the absolute value of the dot product between the normalized score vectors. The generative model using optimal transport appears to have the best alignment since the histogram has a faster decay and a sharper peak at 0 (orthogonality between the score and the leading LV). Although Theorem 4.3 only proves that the orthogonality is a sufficient condition for the robustness of the support, it seems to agree qualitatively with the observations in Figure 5. The most aligned model, being OT-CFM, also exhibits most stability of the predicted support to perturbations. Moreover, none of these models are as robust or as aligned as diffusion models for the same target. These interesting results can open up avenues to pinpoint the most prevalent cause of robustness or lack thereof of the support. Furthermore, our results can be a starting point to understanding the deep connection between the dynamics on sample space that leads to robustness and the dynamics on probability space (which does not uniquely determine the sample space dynamics).

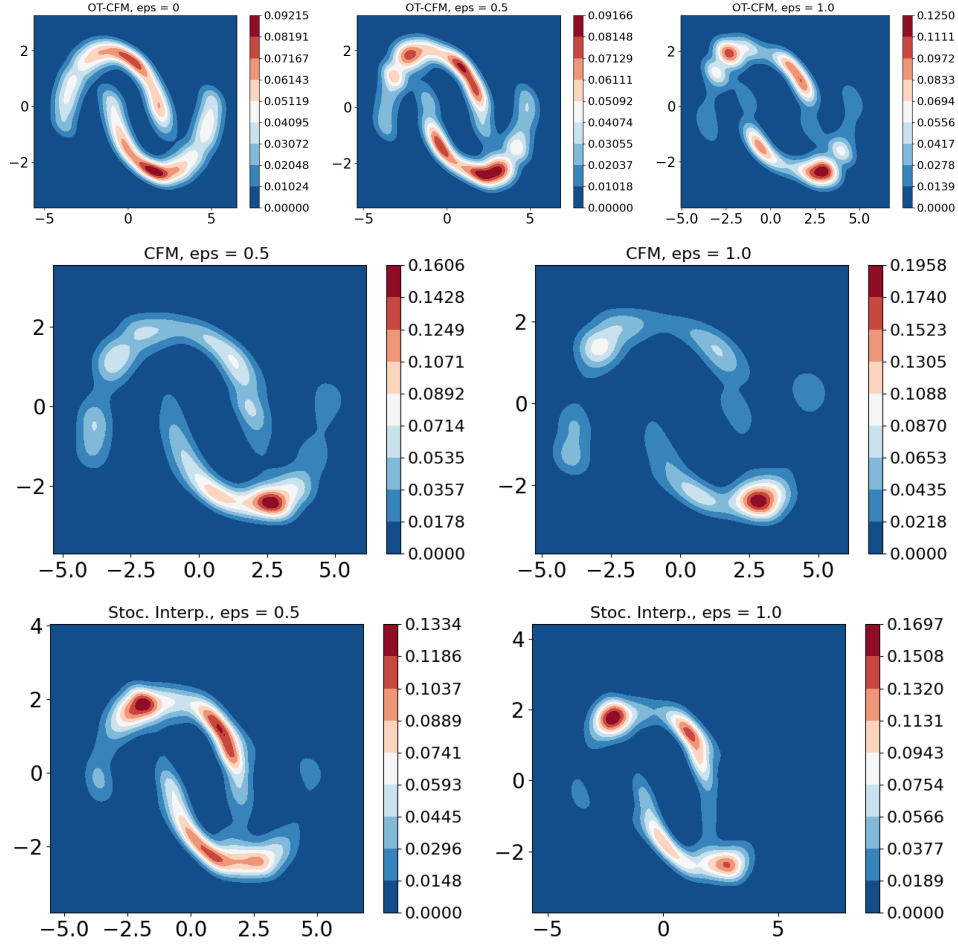


Figure 5: Top row: (Left) Two moons data distribution generated by an optimal transport-conditional flow matching (OT-CFM) algorithm [24]. OT-CFM dynamics perturbed by errors in the vector field of L^∞ norm 0.5 (center) and 1 (right). Middle row: densities predicted by non-rectified flow matching model with perturbations of size 0.5 (left) and 1.0 (right). Bottom: densities predicted by perturbed stochastic interpolant models.

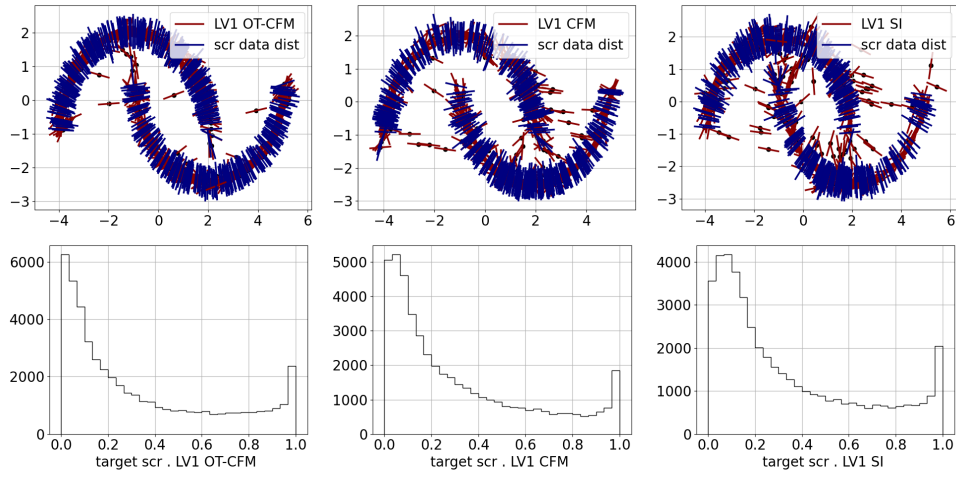


Figure 6: Top row: the target score vector field (blue) and the top LV (red) computed using unperturbed GMs: OT-CFM (left), CFM (center) and stochastic interpolant (right). Bottom row: the histograms of the dot products (absolute value) between the normalized target score and the leading LV (red) over 40,000 points. We see that the stochastic interpolant model and CFM are less aligned than OT-CFM according to our definition in this case.

References

- [1] Michael S Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *arXiv preprint arXiv:2303.08797*, 2023.
- [2] Brian DO Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982.
- [3] Ludwig Arnold, Christopher KRT Jones, Konstantin Mischaikow, Geneviève Raugel, and Ludwig Arnold. *Random dynamical systems*. Springer, 1995.
- [4] Valentin De Bortoli. Convergence of denoising diffusion models under the manifold hypothesis, 2023. URL <https://arxiv.org/abs/2208.05314>.
- [5] Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion schrödinger bridge with applications to score-based generative modeling. *Advances in Neural Information Processing Systems*, 34:17695–17709, 2021.
- [6] Francesco Ginelli, Pietro Poggi, Alessio Turchi, Hugues Chaté, Roberto Livi, and Antonio Politi. Characterizing dynamics with covariant lyapunov vectors. *Physical review letters*, 99(13):130601, 2007.
- [7] Ulrich G Haussmann and Etienne Pardoux. Time reversal of diffusions. *The Annals of Probability*, pages 1188–1205, 1986.
- [8] Desmond J Higham, Xuerong Mao, and Andrew M Stuart. Strong convergence of euler-type methods for nonlinear stochastic differential equations. *SIAM journal on numerical analysis*, 40(3):1041–1063, 2002.
- [9] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. URL <https://arxiv.org/abs/2006.11239>.
- [10] Yuri Kifer. *Ergodic theory of random transformations*, volume 10. Springer Science & Business Media, 2012.
- [11] Hiroshi Kunita. Stochastic differential equations based on lévy processes and stochastic flows of diffeomorphisms. In *Real and Stochastic Analysis: New Perspectives*, pages 305–373. Springer, 2004.
- [12] Hiroshi Kunita and Hiroshi Kunita. *Stochastic flows and stochastic differential equations*, volume 24. Cambridge university press, 1990.
- [13] Pavel V Kuptsov and Ulrich Parlitz. Theory and computation of covariant lyapunov vectors. *Journal of nonlinear science*, 22:727–762, 2012.
- [14] Holden Lee, Jianfeng Lu, and Yixin Tan. Convergence of score-based generative modeling for general data distributions. In *International Conference on Algorithmic Learning Theory*, pages 946–985. PMLR, 2023.
- [15] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=PqvMRDCJT9t>.
- [16] Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=XVjTT1nw5z>.
- [17] Alex Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models, 2021. URL <https://arxiv.org/abs/2102.09672>.
- [18] Bernt Øksendal and Bernt Øksendal. *Stochastic differential equations*. Springer, 2003.
- [19] George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021.

- 399 [20] Bernhard Schölkopf, John C Platt, John Shawe-Taylor, Alex J Smola, and Robert C Williamson.
400 Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–
401 1471, 2001.
- 402 [21] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsuper-
403 vised learning using nonequilibrium thermodynamics. In *International conference on machine*
404 *learning*, pages 2256–2265. pmlr, 2015.
- 405 [22] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and
406 Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv*
407 *preprint arXiv:2011.13456*, 2020.
- 408 [23] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and
409 Ben Poole. Score-based generative modeling through stochastic differential equations. In
410 *International Conference on Learning Representations*, 2021. URL [https://openreview.](https://openreview.net/forum?id=PxtIG12RRHS)
411 [net/forum?id=PxtIG12RRHS](https://openreview.net/forum?id=PxtIG12RRHS).
- 412 [24] Alexander Tong, Kilian FATRAS, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid
413 Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based genera-
414 tive models with minibatch optimal transport. *Transactions on Machine Learning Research*,
415 2024. ISSN 2835-8856. URL <https://openreview.net/forum?id=CD9Snc73AW>. Expert
416 Certification.
- 417 [25] Phil Wang. denoising-diffusion-pytorch. [https://github.com/lucidrains/](https://github.com/lucidrains/denoising-diffusion-pytorch)
418 [denoising-diffusion-pytorch](https://github.com/lucidrains/denoising-diffusion-pytorch), 2024. Accessed: 2025-05-16.
- 419 [26] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang,
420 Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and
421 applications. *ACM Computing Surveys*, 56(4):1–39, 2023.