STABLE GENERATIVE MODELING USING DIFFUSION MAPS

GEORG GOTTWALD, FENGYI LI, YOUSSEF MARZOUK, AND SEBASTIAN REICH

Abstract. We consider the problem of sampling from an unknown distribution for which only a sufficiently large number of training samples are available. Such settings have recently drawn considerable interest in the context of generative modelling. In this paper, we propose a generative model combining diffusion maps and Langevin dynamics. Diffusion maps are used to approximate the drift term from the available training samples, which is then implemented in a discrete-time Langevin sampler to generate new samples. By setting the kernel bandwidth to match the time step size used in the unadjusted Langevin algorithm, our method effectively circumvents any stability issues typically associated with time-stepping stiff stochastic differential equations. More precisely, we introduce a novel split-step scheme, ensuring that the generated samples remain within the convex hull of the training samples. Our framework can be naturally extended to generate conditional samples. We demonstrate the performance of our proposed scheme through experiments on synthetic datasets with increasing dimensions and on a stochastic subgrid-scale parametrization conditional sampling problem.

1. Introduction

Generative modeling is the process of learning a mechanism for synthesizing *new* samples that resemble those of the original data-generating distribution, given only a finite set of samples. It has seen wide adoption and enormous success across diverse application domains, from image [5, 21, 43, 49] and text generation [63, 29, 30], to drug discovery [3, 2] and anomaly detection [10, 47], to name but a few.

In this paper, we introduce a new nonparametric approach to generative modeling that combines ideas from optimal transportation, diffusion maps, and Langevin dynamics.

Suppose that we are given M training samples $x^{(i)} \sim \pi$, $i = 1, \dots, M$, from an unknown distribution π on $\mathbb{R}^d$. Perhaps the simplest nonparametric approach to generative modeling is to build a kernel density estimate (KDE) and then sample from it; the KDE is essentially a mixture model with M components. Alternatively, one could estimate the score function, $\hat{s}_M(x) \approx s(x) := \nabla \log \pi(x)$, without directly estimating π, and use this estimate as the drift term of Langevin dynamics,

$$\dot{X}_t = \hat{s}_M(X_t) + \sqrt{2}\dot{W}_t, \tag{1}$$

where W_t denotes standard d-dimensional Brownian motion.

There are myriad ways of estimating the score function [22, 54], and given an estimate for it, one needs to discretize (1), for example using Euler–Maruyama, to obtain an implementable scheme. However, the step size needs to be carefully chosen: a small step size leads to slow convergence, while too large a step yields instability of the numerical scheme, especially for data that are supported on a compact manifold, e.g.,

$$\mathcal{M} = \{x \in \mathbb{R}^d : g(x) = 0\}, \tag{2}$$

Date: January 9, 2024.

for some unknown function $g(x)$. Any estimated score function $\hat{s}_M(x)$ will take large values for x with $\|g(x)\|^2 \gg 0$, rendering the Langevin dynamics (1) stiff. In other words, an explicit time integration method such as Euler–Maruyama will require extremely small step sizes.

In this paper, we choose an alternative approach. Instead of first estimating the score function $\hat{s}_M(x)$ and then discretising (1) in time, we employ diffusion maps [8, 7, 37] to directly approximate the semigroup $\exp(\epsilon\mathcal{L})$, $\epsilon > 0$, of a diffusion process with generator $\mathcal{L}$ and invariant distribution π, from the given samples $\{x^{(i)}\}_{i=1}^M$. The second key ingredient of our method is to interpret ϵ as a step size and to read off a Gaussian transition kernel from the diffusion map in the form of

$$X_{n+1} = m_\epsilon(X_n) + \Sigma(X_n)\Xi_n, \tag{3}$$

with appropriately defined functions $m_\epsilon : \mathbb{R}^d \to \mathbb{R}^d$ and $\Sigma : \mathbb{R}^d \to \mathbb{R}^{d\times d}$, and $\Xi_n \sim \mathrm{N}(0, I)$. Broadly, $m_\epsilon(X_n)$ controls the drift, while $\Sigma(X_n)\Xi_n$ introduces noise. In this recursion, the step size ϵ is linked with the error of the diffusion map approximation, making the time discretization implicit. Comparing to directly discretizing (1) using Euler–Maruyama with step-size $\Delta t = \epsilon$, we will demonstrate that the scheme (3) is stable and ergodic for all step-sizes $\epsilon > 0$ and, hence, ϵ can be chosen solely on accuracy considerations. Furthermore, the inclusion of the position-dependent diffusion matrix $\Sigma(x)$ makes it better suited for sampling from a manifold.

While our approach is rooted in diffusion maps [8, 7, 37, 38], our construction utilizes a Sinkhorn scheme [58], which is motivated by considering a special case of the discrete Schrödinger bridge problem of optimally coupling the empirical measure of the training samples with itself. By solving the Schrödinger bridge problem, we construct a transition matrix whose state space encompasses all the training points. We generalize this approach to the continuous state space that extends beyond the current training data points. Armed with this transition kernel, we obtain a Markov chain that samples from the underlying distribution of the training data via (3). From here, we introduce a novel split-step time-stepping scheme, which ensures that the generated samples consistently lie within the convex hull of the training samples. In contrast, using a direct discretization of (1) with Euler–Maruyama results in generating samples on unbounded domains.

In addition, we replace the fixed bandwidth kernel with a variable bandwidth kernel within our diffusion map framework. As we demonstrate in our numerical experiments, the resulting sampling scheme (3) provides a better representation of the underlying target distribution. More precisely, we assess the quality of the generated samples using a variable bandwidth kernel and using a fixed bandwidth kernel on synthetic data sets drawn from non-uniform distributions supported on irregular domains and on low-dimensional manifolds.

We then extend our method to create a conditional generative model. This allows to perform Bayesian inference in the "simulation-based" setting, i.e., without explicit evaluation of a prior density or likelihood. We demonstrate the performance of this approach in a stochastic subgrid-scale parametrization problem.

1.1. **Related work.** Langevin dynamics (1) characterizes the motion of particles as they experience a blend of deterministic and stochastic forces. Unlike in this paper, it is typically assumed that the deterministic forcing term $\nabla \log \pi(x)$ is given. Langevin dynamics has been used as a popular tool for sampling data from the target distribution π. One variation of this is to introduce a symmetric preconditioning operator to the Langevin dynamics. Some popular choices of the preconditioning include the empirical covariance [12, 6, 39] and the Riemannian metric [13, 61, 28], making this method

converge faster and more geometry-aware, while leaving the stationary distribution unchanged.

On the other hand, diffusion maps have traditionally served as a tool for nonlinear dimensional reduction [8, 7, 37]. The kernel matrix formed using pairwise sample distances with appropriate normalization approximates the semigroup $\exp(\epsilon\mathcal{L})$. In recent work [58], an alternative normalization technique based on Sinkhorn weights has been studied. The Sinkhorn algorithm solves for the Markov transition kernel associated with a discrete Schrödinger bridge problem, where the coupling is between the empirical measures of the training samples with themselves. This approach results in a *symmetric* stochastic operator that, notably, also approximates the semigroup $\exp(\epsilon\mathcal{L})$. The drift term of the Langevin dynamics is estimated by acting on the identity function. Separately, the idea of using variable bandwidth kernels can be found very early in the statistics community for kernel density estimation [45, 56], for estimating regression curves [36] and mean regression functions [11]. Recently, [1] replaces the original fixed bandwidth kernel with the variable bandwidth kernel in the construction of diffusion maps, making the approximation of the generator accurate on *unbounded* domains. Inspired by this concept, we replace the fixed bandwidth kernel with a variable bandwidth kernel. The resulting normalized matrix approximates the semigroup of a different Langevin diffusion process.

In recent years, there has been a surge of research interest in the realm of generative modeling. Despite the able achievements of well-established neural network-based generative models, such as variational auto-encoders (VAE) [25, 42], generative adversarial networks (GAN) [14], and diffusion models or score generative models (SGM) [20, 51, 55, 62], they often require meticulous hyperparameter tuning [44, 53] and exhibit a long training time [14, 57, 50]. The efficacy of these methods significantly hinges on the architectural choices and parameter settings of the underlying deep neural networks [46, 24], which, regrettably, demands a high level of expertise. Furthermore, SGMs solve both a forward and a reverse stochastic differential equation (SDE). The forward SDE introduces noise to the sample, transforming the data into the standard normal distribution, while the reverse SDE takes sample from the standard normal distribution back to the original data distribution, yielding a different sample than the one initially fed into the forward SDE. During the training process, the score function is learned, not for the target distribution, but for the data distribution at each time. Our work, on the other hand, solves only one (forward) SDE, and we learned the score of the target distribution only once.

Several recent studies have combined a range of score function estimation techniques with Langevin dynamics. For example, [52] introduces a noise conditional score network to learn the score function and then uses annealed Langevin dynamics to generate samples, and [4] studies the convergence rate of a Langevin based generative model, where the score is estimated using denoising auto-encoders. Such techniques are also studied within the Bayesian imaging community, commonly referred to as "plug and play" [27]. Nevertheless, these approaches use neural networks for the estimation of the score function, necessitating substantial fine-tuning, and their effectiveness depends on factors such as the complexity of the approximation families and the architectural structures of the neural networks. In addition, [27] uses an explicit projection to ensure that samples stay on the compact manifold given by (2) which is assumed to be explicitly known. In contrast, we do not assume any knowledge of $\mathcal{M}$.

1.2. **Outline.** In Section 2 we construct a Markov chain using a Schrödinger bridge diffusion map approximation that samples from the given discrete data distribution. In Section 3, we extend this Markov chain to the continuous setting by constructing a

Gaussian transition kernel which extracts its conditional mean and covariance matrix from the underlying diffusion map approximation. We introduce two discrete-time Langevin samplers; one with an arbitrary data-unaware diffusion and one with a data-aware diffusion matrix in Section 3.1. Theoretical properties such as stability and ergodicity are discussed in Section 3.2. We further discuss the application of variable bandwidth kernels when estimating the diffusion map in Section 3.3. While Section 3 focuses on finite step-size and finite sample-size implementations, Section 4 establishes connections to the underlying semi-groups and generators in the infinite sample-size limit. We explore the extension of our proposed scheme to a conditional sampling setting in Section 5, and we demonstrate our proposed methods in Section 6 in a suite of examples, including a conditional sampling exercise with an application to stochastic subgrid-scale parametrization. We conclude in Section 7 with a summary and an outlook.

2. Discrete Schrödinger bridges

In this section, we collect some preliminary building blocks by considering the simpler task of building a discrete Markov chain over the samples $\{x^{(i)}\}_{i=1}^M$, which leaves the associated empirical probability measure

$$\mu_{\rm em}(\mathrm{d}x) = \frac{1}{M}\sum_{i=1}^M \delta_{x^{(i)}}(\mathrm{d}x)$$

in $\mathbb{R}^d$ invariant. Here $\delta_x(\mathrm{d}x)$ denotes the Dirac delta distribution centred at x. In the subsequent section, we will generalise the finding from this section to approximately sample from π, allowing for the generation of new samples which are different from the given training samples.

We consider the Schrödinger bridge problem of coupling $\mu_{\rm em}$ with itself along a reversible reference process of the form

$$X' = X + \sqrt{\epsilon}\left(K(X') + K(X)\right)^{1/2}\Xi, \tag{4}$$

where $\epsilon > 0$ is a tuneable parameter, $\Xi \sim \mathrm{N}(0, I)$, and $K(x)$ is a symmetric positive definite matrix for all $x \in \mathbb{R}^d$. Popular choices include $K = I$, $K = \Sigma_M$, where Σ_M is the empirical covariance matrix of the samples $\{x^{(i)}\}_{i=1}^M$, and $K = \rho(x)I$, where $\rho(x) > 0$ is a scaling function representing variable bandwidth. The update step (4) corresponds to a Stratonovitch SDE with multiplicative noise and diffusion matrix $K(x)$. This provides a natural way of interpreting $K(X') + K(X)$ as a preconditioner for the associated Schrödinger bridge problem, and this fact will be explored in the context of variable bandwidth implementations in Section 3.3.

Instead of working with the empirical measure $\mu_{\rm em}(\mathrm{d}x)$, we introduce the probability vector $p^* = (1/M, \ldots, 1/M)^{\rm T} \in \mathbb{R}^M$ over $\{x^{(i)}\}_{i=1}^M$. Then the Schrödinger bridge problem is solved by first introducing the symmetric matrix $T_\epsilon \in \mathbb{R}^{M\times M}$ with entries

$$t_{ij} = \exp\left(-\frac{1}{2\epsilon}(x^{(i)} - x^{(j)})^{\rm T}\left(K(x^{(i)}) + K(x^{(j)})\right)^{-1}(x^{(i)} - x^{(j)})\right). \tag{5}$$

One then finds the scaling vector $v_\epsilon \in \mathbb{R}^M$ such that the symmetric matrix

$$P_\epsilon = D(v_\epsilon)T_\epsilon D(v_\epsilon) \tag{6}$$

satisfies

$$P_\epsilon 1_M = p^*.$$

Here $1_M = (1, \ldots, 1)^{\rm T} \in \mathbb{R}^M$, and $D(v) \in \mathbb{R}^{M\times M}$ denotes the diagonal matrix with diagonal entries provided by $v \in \mathbb{R}^M$.

Given P_ϵ, one can now construct a Markov chain that samples from μ_{em}. Assume the Markov chain is currently in state $x^{(j)}$, then the transition probabilities to the next state $x \in \{x^{(i)}\}_{i=1}^M$ are given by

$$p_j = MP_\epsilon e_j \in \mathbb{R}^M,$$

where $e_j \in \mathbb{R}^M$ denotes the j-th unit vector in $\mathbb{R}^M$. Since all entries in P_ϵ are bounded from below provided all samples satisfy $x^{(i)} \in \mathcal{M}$, where $\mathcal{M}$ is a compact submanifold in $\mathbb{R}^d$, the constructed Markov chain possesses a unique invariant measure given by p^* and is geometrically ergodic. The rate of convergence can be determined by the diffusion distance

$$d(x^{(i)}, x^{(j)}) = \|p_i - p_j\|^2.$$

If the diffusion distance is small, then $x^{(i)}$ and $x^{(j)}$ are well connected. Furthermore, if $d(x^{(i)}, x^{(j)})$ is small for all points, then the Markov chain will mix quickly. In particular, larger values of ϵ will lead to faster mixing. This idea can be further extended to the case where the samples are not uniformly weighted. In Bayesian inference, for example, the weight at each data point is proportional to its likelihood. We illustrate this idea in the following remark.

Remark 2.1. *If there is a change of measure due to, for example, observed data with likelihood $\pi(y|x)$, then the resulting new empirical measure is given by*

$$\hat{\mu}_{\text{em}}(\mathrm{d}x) = C^{-1} \sum_{i=1}^M \pi(y|x^{(i)}) \delta_{x^{(i)}}(\mathrm{d}x),$$

where $C = \sum_{i=1}^M \pi(y|x^{(i)})$. Let us introduce the probability vector

$$\hat{p}^* = C^{-1}(\pi(y|x^{(1)}), \ldots, \pi(y|x^{(M)}))^{\mathrm{T}} \in \mathbb{R}^M.$$

The associated coupling is now provided by

$$\hat{P}_\epsilon = D(\hat{v}_\epsilon) T_\epsilon D(\hat{v}_\epsilon)$$

subject to

$$\hat{P}_\epsilon 1_M = \hat{p}^*.$$

Furthermore, the transition probabilities $p_j \in \mathbb{R}^M$, $j = 1, \ldots, M$, get replaced by

$$\hat{p}_j = \frac{\hat{P}_\epsilon e_j}{1_M^{\mathrm{T}} \hat{P}_\epsilon e_j} \in \mathbb{R}^M$$

and the associated Markov chain will sample from the empirical posterior distribution $\hat{\mu}_{\text{em}}(\mathrm{d}x)$.

However, the goal is to approximately sample from the underlying distribution π and not just the empirical distribution $\mu_{\text{em}}(\mathrm{d}x)$. The required extension of our baseline algorithm is discussed in the following section.

3. Diffusion approximation

In order to implement (3), we need to define $m_\epsilon(x)$ and $\Sigma(x)$ for $x \in \mathbb{R}^d$. In this section, we discuss how one can obtain these functions from the training samples $\{x^{(i)}\}_{i=1}^M$ and the diffusion map approximation (6).

Given the underlying reference process (4) and samples $\{x^{(i)}\}_{i=1}^M$, recall that $X'|X = x \sim N(x, \epsilon(K(x)+K(X')))$. We therefore introduce the vector $t_\epsilon(x) \in \mathbb{R}^M$ with entries

$$t_{\epsilon,i}(x) = \exp\left(-\frac{1}{2\epsilon}(x^{(i)} - x)^{\mathrm{T}} \left(K(x) + K(x^{(i)})\right)^{-1} (x^{(i)} - x)\right) \tag{7}$$

for $i = 1, \ldots, M$. We then define the probability vector using the Sinkhorn weights, v_ϵ, obtained in (6), i.e.,

$$p_\epsilon(x) = \frac{D(v_\epsilon) t_\epsilon(x)}{v_\epsilon^{\mathrm{T}} t_\epsilon(x)} \tag{8}$$

for all $x \in \mathbb{R}^d$. This vector gives the transition probabilities from x to $\{x^{(i)}\}_{i=1}^M$ and provides a finite-dimensional approximation to the conditional probability distribution $\pi_\epsilon(\cdot|x)$ of the true underlying diffusion process; that is, the semigroup $\exp(\epsilon\mathcal{L})$ with generator $\mathcal{L}$. See Section 4 for more details. In the following, we further approximate $\pi_\epsilon(\cdot|x)$ by a Gaussian and estimate its mean and covariance matrix using the probability vector p_ϵ and the data matrix of samples

$$\mathcal{X} = (x^{(1)}, \ldots, x^{(M)}) \in \mathbb{R}^{d \times M}. \tag{9}$$

3.1. **Sampling algorithms.** We now present our main Langevin sampling strategies based on the diffusion map approximation for the probability vector $p_\epsilon(x)$ in (8). Our sampling schemes have the same drift term, $m_\epsilon(x)$, but differ in the way the diffusion matrix, $\Sigma(x)$, is defined. We consider a data-unaware diffusion as well as a data-aware diffusion which turns out to be advantageous in generating new samples from the data distribution π. See the numerical experiments in Section 6.

3.1.1. *Langevin sampler with data-unaware diffusion.* Following the proposed methodology, we introduce the sample-based approximation of the conditional mean

$$m_\epsilon(x) := \mathcal{X} p_\epsilon(x). \tag{10}$$

Remark 3.1. *The construction of the conditional mean $m_\epsilon(x)$ is known as the barycentric projection of the entropy-optimally coupling [48, 40]. In optimal transport, v_ϵ plays the role of the optimizer of the dual problem.*

Using $m_\epsilon(x)$, we propose the recursive sampler

$$X_{n+1} = X_n + \Delta\tau \left(\frac{m_\epsilon(X_n) - X_n}{\epsilon} \right) + \sqrt{2\Delta\tau} K(X_n)^{1/2} \Xi_n, \tag{11}$$

where $\Delta\tau$ is the time step. In other words, we obtain the score function approximation

$$s_M(x) = \frac{m_\epsilon(x) - x}{\epsilon} \tag{12}$$

in (1). By taking $\Delta\tau = \epsilon$, we have

$$X_{n+1} = m_\epsilon(X_n) + \sqrt{2\epsilon} K(X_n)^{1/2} \Xi_n. \tag{13}$$

Note that (13) fits into the general formulation (3) with $\Sigma(x) = \sqrt{2\epsilon} K(x)^{1/2}$.

The parameter $\epsilon > 0$ can be seen as a step-size. For large ϵ, the expected value $m_\epsilon(x)$ will become essentially independent of the current state X_n and the diffusion process will sample from a centred Gaussian. For $\epsilon \to 0$, on the other hand, the probability vector $p_\epsilon(x)$ can potentially degenerate into a vector with a single entry approaching one with all other entries essentially becoming zero. Hence a key algorithmic challenge is to find a good value for ϵ and a suitable $K(x)$, which guarantee both good mixing and accuracy, that is, $X_n \sim \pi$ as $n \to \infty$.

3.1.2. *Langevin sampler with data-aware diffusion.* From (8) and (9), one can also define the conditional covariance matrix,

$$C(x) = (\mathcal{X} - m_\epsilon(x)1_{\mathrm{M}}^{\mathrm{T}})D(p_\epsilon(x))(\mathcal{X} - m_\epsilon(x)1_{\mathrm{M}}^{\mathrm{T}})^{\mathrm{T}} \in \mathbb{R}^{d\times d}, \tag{14}$$

which is the covariance matrix associated with the probability vector $p_\epsilon(x)$. Therefore, one can more directly implement a Gaussian approximation associated with the transition probabilities $p_\epsilon(x)$ and introduce the update

$$X_{n+1} = X_n + \Delta\tau\left(\frac{m_\epsilon(X_n) - X_n}{\epsilon}\right) + \sqrt{\Delta\tau/\epsilon}C(X_n)^{1/2}\Xi_n. \tag{15}$$

Similar to the previous case, setting $\Delta\tau = \epsilon$ implies

$$X_{n+1} = m_\epsilon(X_n) + C(X_n)^{1/2}\Xi_n, \tag{16}$$

which we found to work rather well in our numerical experiments since it directly captures the uncertainty contained in the data-driven coupling P_ϵ. The scheme (16) corresponds to setting $\Sigma(x) = C(x)^{1/2}$ in (3). Also note that the schemes (16) still depends on $K(X)$ through the probability vector $p_\epsilon(x)$.

It is not always justifiable to use $\Delta\tau = \epsilon$ as a step-size and $\Delta\tau < \epsilon$ can be beneficial instead. In those cases, one can resort to the re-scaled time-stepping methods (11) or (15), respectively.

3.2. **Algorithmic properties.** We briefly discuss several important properties on the stability and the ergodicity of the proposed Langevin samplers.

The following Lemma establishes that, since each $p_\epsilon(x)$ is a probability vector, $m_\epsilon(x) = \mathcal{X}p_\epsilon(x)$ is a convex combination of the training sample $\{x^{(i)}\}_{i=1}^M$.

Lemma 3.1. *Let us denote the convex hull generated by the data points $\{x^{(i)}\}_{i=1}^M$ by $\mathcal{C}_M$. It holds that*

$$m_\epsilon(x) \in \mathcal{C}_M \tag{17}$$

for all choices of $\epsilon > 0$ and all $x \in \mathbb{R}^d$.

Proof. Follows from the definition (10) and the fact that $p_\epsilon(x)$ is a probability vector for all $\epsilon > 0$ and all $x \in \mathbb{R}^d$. □

This establishes stability of the Langevin samplers (13) and (16) for all step-sizes $\epsilon > 0$.

The next lemma shows that the Langevin sampler (13) is geometrically ergodic toward the invariant measure π.

Lemma 3.2. *Let us assume that the data generating density π has compact support. Then the time-stepping method* (13) *possesses a unique invariant measure and is geometrically ergodic provided the norm of the symmetric positive matrix $K(x)$ is bounded from above and below for all $x \in \mathbb{R}^d$.*

Proof. Consider the Lyapunov function $V(x) = 1 + \|x\|^2$ and introduce the set

$$C = \{x : \|x\| \le R\}$$

for suitable $R > 0$. Since $m_\epsilon(X_n) \in \mathcal{C}_M$ and π has compact support, one can find a radius $R > 0$, which is independent of the training data $\{x^{(i)}\}$, such that $\mathcal{C}_M \subset C$ and

$$\mathbb{E}[V(X_{n+1})|X_n] \le \lambda V(X_n)$$

for all $X_n \notin C$ with $0 \le \lambda < 1$. Furthermore, because of the additive Gaussian noise in (13), there is a probability density function $\nu(x)$ and a constant $\delta > 0$ such that

$$\mathrm{n}(x'; m_\epsilon(x), 2\epsilon K(x)) \ge \delta\nu(x')$$

for all $x, x' \in C$. Here $\mathrm{n}(x; m, \Sigma)$ denotes the Gaussian probability density function with mean m and covariance matrix Σ. In other words, C is a small set in the sense of [35]. Geometric ergodicity follows from Theorem 15.0.1 in [35]. See also the self-contained presentation in [33]. □

Extending Lemma 3.2 to the time-stepping scheme (16) is non-trivial since the covariance matrix (14) may become singular.

Lemma 3.1 also suggests to replace the sampling step (13) by the associated split-step scheme

$$X_{n+1/2} = X_n + \sqrt{2\epsilon} K(X_n)^{1/2} \Xi_n, \tag{18a}$$

$$X_{n+1} = m_\epsilon(X_{n+1/2}). \tag{18b}$$

This scheme now satisfies $X_n \in \mathcal{C}_M$ for all $n \geq 1$ and any choice of ϵ. Similarly, one can replace (16) by the split-step scheme

$$X_{n+1/2} = X_n + C(X_n)^{1/2} \Xi_n, \tag{19a}$$

$$X_{n+1} = m(X_{n+1/2}). \tag{19b}$$

These split-step schemes have been used in our numerical experiments.

3.3. **Variable bandwidth diffusion.** It is well-known from the literature on diffusion maps that a variable bandwidth can improve the approximation quality for fixed sample size M [1]. Here we utilize the same idea. However, we no longer insist on approximating the standard generator with $K = I$, since we only wish to sample from the distribution π rapidly. Hence, we consider (4) with $K(x)$ of the form

$$K(x) = \rho(x) I. \tag{20}$$

It is an active area of research to select a ρ that increases the spectral gap of $\mathcal{L}$ while not increasing computational complexity. Indeed, a larger spectral gap implies a faster convergence rate [41], indicating that the generated samples are closer to the reference at a finite time, exhibiting a high accuracy. We demonstrate numerically in Section 6 that ρ can indeed be used to increase the sampling accuracy. More specifically, the bandwidth $\rho(x)$ is chosen as

$$\rho(x) = \pi(x)^\beta, \tag{21}$$

where $\beta \leq 0$ is a parameter and the unknown sampling distribution π is approximated by an inexpensive low accuracy density estimator. One finds that the variable bandwidth parameter β in (21) and the scaling parameter ϵ both influence the effective step-size in the reference process (4) for (20). In order to disentangle the two scaling effects we modify the construction of the entries (5) in T_ϵ as follows. We first compute $\pi_i \approx \pi(x^{(i)})$ over all data points and then rescale these typically unnormalized densities:

$$\tilde{\pi}_i = Z^{-1} \pi_i, \qquad Z := \frac{1}{M} \sum_j \pi_j.$$

The variable scaling length is then set to

$$\rho_i = \tilde{\pi}_i^\beta = Z^{-\beta} \pi_i^\beta$$

for $i = 1, \ldots, M$, that is, $K(x^{(i)}) = \rho_i I$ in (5) and, more generally,

$$K(x) = Z^{-\beta} \pi(x)^\beta. \tag{22}$$

The proposed scaling implies that a constant target density $\pi(x)$ leads to $K(x^{(i)}) = I$ in (5) regardless of the chosen β value. See Section 6 below for our numerical findings.

4. Connections to the generator of Langevin dynamics

In this section, we discuss the proposed methodology in the limit $M \to \infty$ under the idealised assumption that the target distribution π is in fact known. It is well-known that $K = I$ implies that the diffusion map approximates the semi-group corresponding to the generator of standard Langevin dynamics. Employing a variable bandwidth (20) instead, i.e. $K = \rho(x)I$, alters the semi-group and thus the underlying generator. However, since in both cases the density $\pi(x)$ remains invariant, this is not an issue as we are only concerned with drawing samples from the distribution and not in reproducing any dynamical features.

More precisely, using (20) in (4), we consider the diffusion map approximation given by (6). We formally take the limit $M \to \infty$ and denote the limiting operator by $\mathcal{P}_\epsilon$ [58]. One formally obtains

$$\mathcal{P}_\epsilon = e^{\epsilon \mathcal{L}},$$

where $\mathcal{L}$ denotes the generator defined by

$$\mathcal{L} f = \pi^{-1} \nabla \cdot (\pi \rho \nabla f) = \nabla \cdot (\rho \nabla f) + \rho \nabla \log \pi \cdot \nabla f \tag{23}$$

We note that $\mathcal{L}$ is self-adjoint (reversible) with respect to the π-weighted inner product. It is more revealing to consider the dual operator

$$\mathcal{L}^\dagger \mu = \nabla \cdot (\mu \rho \nabla (\log \mu - \log \pi))$$

and the associated mean-field evolution equation

$$\dot{X}_t = \rho(X_t) \left(\nabla (\log \pi(X_t) - \log \mu(X_t)) \right), \qquad X_0 \sim \mu_0,$$

which implies the invariance of $\mu = \pi$.

Please also note that the generator corresponds to the diffusion process

$$\dot{X}_t = \rho(X_t) \nabla \log \pi(X_t) + \nabla \rho(X_t) + \sqrt{2\rho(X_t)} \dot{W} \tag{24}$$

in Itô form and to

$$\dot{X}_t = \rho(X_t) \nabla \log \pi(X_t) + \sqrt{2\rho(X_t)} \circ \dot{W}$$

in Stratonovitch form. The drift term in the Itô formulation (24) can be expressed as

$$\mathcal{L}\,\mathrm{Id} = \rho \nabla \log \pi + \nabla \rho \tag{25}$$

and, hence, the score function (12) is a finite M approximation to

$$s(x) = \mathcal{L}\,\mathrm{Id}(x).$$

However, recall that the recursive Langevin sampler (13) relies on a direct time-stepping approach to (24) and is based on

$$m_\epsilon(x) = \mathcal{X} p_\epsilon(x) \approx \exp(\epsilon \mathcal{L})\,\mathrm{Id}(x)$$

instead of an approximation of the drift term on the right hand side of (24) via (25) followed by Euler–Maruyama. However, both approaches are formally consistent in the limit $\epsilon \to 0$ as

$$\exp(\epsilon \mathcal{L})\,\mathrm{Id}(x) \approx x + \epsilon s(x).$$

While we have focused on the particular choice $K(x) = \rho(x)I$ in this section for simplicity, the discussion naturally extends to symmetric positive definite matrices $K(x)$.

5. Conditional sampling

In this section, we explore an extension of the sampling scheme (18) to the context of conditional generative modeling. More specifically, consider a random variable in $x = (y, z)$, which we condition on the second component for given $z = z^*$. In other words, we wish to sample from $\pi(y|z^*)$ given samples $x^{(i)} = (y^{(i)}, z^{(i)})$, $i = 1, \ldots, M$, from the joint distribution $\pi(x) = \pi(y, z)$.

Bayesian inference arises as a popular application of conditional sampling. In the traditional Bayesian inference framework, to generate samples from the posterior distribution, one typically requires access to both the prior and the likelihood, namely,

$$\hat{\pi}(y) := \pi(y|z^*) \propto \pi(z^*|y)\, \pi(y).$$

However, conditional sampling provides a way to sample from the posterior just via the samples from the joint, obviating the need for the Bayesian update. In order to perform the required conditional sampling, we propose a method which combines approximate Bayesian computation (ABC) with our diffusion map based sampling algorithm.

Let us assume that we can generate M samples $x^{(i)} = (y^{(i)}, z^{(i)})$, $i = 1, \ldots, M$, from a joint distribution $\pi(y, z)$, which we then wish to condition on a fixed z^*. As before, we construct vectors of transition probabilities $p_\epsilon(x) \in \mathbb{R}^M$ based on the samples $\{x^{(i)}\}_{i=1}^M$. We assume that the bandwidth parameter ϵ used in the diffusion map approximation is also applied in the ABC misfit function, that is,

$$L(z, z^*) = \frac{1}{2\epsilon}\|z - z^*\|^2.$$

This suggests the following split-step approximation scheme. Given the last sample $X_n = (Y_n, Z_n)$, we first update the z-component using

$$\hat{Z}_n = Z_n - \epsilon \nabla_z L(Z_n, z^*) = z^*.$$

In other words, we replace the current X_n by $\hat{X}_n = (Y_n, z^*)$. Next we apply the split-step scheme (18) to $\hat{X}_n$, that is,

$$X_{n+1/2} = \hat{X}_n + \sqrt{2\epsilon} K(\hat{X}_n)^{1/2} \Xi_n,$$

and

$$X_{n+1} = m_\epsilon(X_{n+1/2}) = \mathcal{X} p_\epsilon(X_{n+1/2})$$

where $\mathcal{X} = (x^{(1)}, \ldots, x^{(M)})$ and the definition of the probability vectors $p_\epsilon(x)$ follows from (8). The split-step scheme (19) generalises along the same lines.

6. Numerical experiments

In this section, we illustrate our method through three numerical examples encompassing different ranges and focal points. In the first two examples,we generate samples using synthetic datasets with increasing dimensions. Our emphasis is on exploring the impact of employing the variable bandwidth kernel. In the third example, we showcase the proposed conditional generative modeling in Section 5, applied to a stochastic subgridscale parametrization problem.

6.1. **One-dimensional manifold.** To illustrate how well the proposed methods generate statistically reliable samples we consider first the case of M samples $x \in \mathbb{R}^2$. In particular, we consider samples with a polar representation with radius $r = 1 + \sigma_r \xi_r$ and angle $\theta = \pi/4 + \sigma_\theta \xi_\theta$ with $\sigma_r = 0.06$ and $\sigma_\theta = 0.6$ and $\xi_{r,\theta} \sim \mathrm{N}(0, 1)$. We used $M = 2,000$ samples to learn the transition kernel and then generated $10,000$ new

samples with an initial condition at the tail with the data point corresponding to the smallest angle.

We begin by investigating the effect of the two noise models proposed, namely a constant diffusion as in (13) with constant bandwidth $K(x) = I$ and the case when the diffusion reproduces the sample covariance C as in (16). We employ a Langevin sampler with the splitting scheme (18) with $\epsilon = 0.009$ and (19), respectively. Figure 1 shows that choosing the sample covariance as the noise model is clearly advantageous. Whereas both noise models generate samples that reproduce the angular distribution the noise model using a constant diffusion is overdiffusive in the radial direction. In contrast the noise model using the sample covariance nicely reproduces the radial distribution.

We now investigate the effect of a variable bandwidth $K(x)$. We employ the noise model (19) with the sample covariance but use $K(x)$ to determine the diffusion map (cf. (7)). The Langevin sampler (19) is again initialized with the coordinates of the data point corresponding to the smallest angle in the data-sparse tail. Figure 2a shows the samples projected onto the convex hull of the data, i.e. outputs of step (19b), when a uniform bandwidth $K(x) = I$ with $\epsilon = 0.009$ is employed. Although the mean behaviour is well reproduced, it is seen that the generative model fails near the data-sparse tails for large and small angles. Here the value of ϵ is too small to generate significant diffusion and the samples are aligned on the (linear) convex hull of the widely separated data samples. To mitigate against this behaviour we employ a variable bandwidth $\rho(x) = \pi^{\beta}$ with $\beta = -1/5$ and a kernel density estimate $\pi(x)$. Figure 2b shows how the variable bandwidth kernel is able to better reproduce the sampling in the data-sparse tail regions. Figure 3 shows the empirical histograms for the radius and the angle variables of the noisy samples corresponding to Figure 2 (i.e. outputs of step (19a)). Whereas both, the uniform and the variable bandwidth kernels, reproduce the radial distribution very well, the uniform bandwidth fails to reproduce the angular distribution in the tails where the diffusion is not sufficiently strong to let the sampler escape the convex hull of the data.

We have seen that a constant uniform bandwidth generates samples which are concentrated in the bulk of the data and which are overly diffusive in the radial direction (cf. Figure 1c). One may wonder if employing a smaller virtual time step $\Delta\tau < \epsilon$ in the Langevin sampler (11) will allow a Langevin sampler with constant bandwidth to generate more faithful samples. Figure 4 shows that choosing a smaller time step $\Delta\tau$ in (15), here with $\Delta\tau = \epsilon/4$ is indeed able to reproduce the radial distribution. However, if the Langevin sampler is initialised with a data point in the center of the data samples, it is not able to diffuse into the tail of the distribution distribution, leading to an under-diffusive empirical histogram for the angles.

The numerical experiments above suggest that we employ a noise model using the sample covariance C combined with a variable bandwidth $K(x)$ to control eventual data sparse regions. When employing a variable bandwidth our method contains two hyperparameters which require tuning: the bandwidth factor ϵ and the exponent β in the arbitrary choice of the variable bandwidth $\rho(x)$. Their role will be explored in the following subsection.

6.2. **Multi-dimensional manifolds.** In this numerical example, we show our proposed method on hyper semi-spheres of dimension $d = \{3, 4, 9\}$, using both a fixed bandwidth kernel and a variable bandwidth kernel. Data are generated by firstly sampling $z^{(i)} = (z_1^{(i)}, \cdots, z_d^{(i)})$ from a d-dimensional standard normal distribution, and

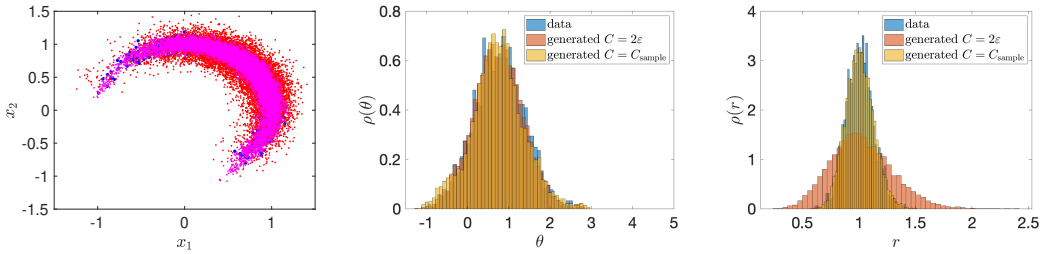

FIGURE 1. Comparison of the different noise models employed by the generative model. We employed a constant bandwidth with $\epsilon = 0.009$. Left: Original (blue) and generated data using a constant covariance (red) and the sample covariance $C(x)$ (magenta). Middle: Empirical histograms of the angular variable θ. Right: Empirical histograms of the radial variable r.

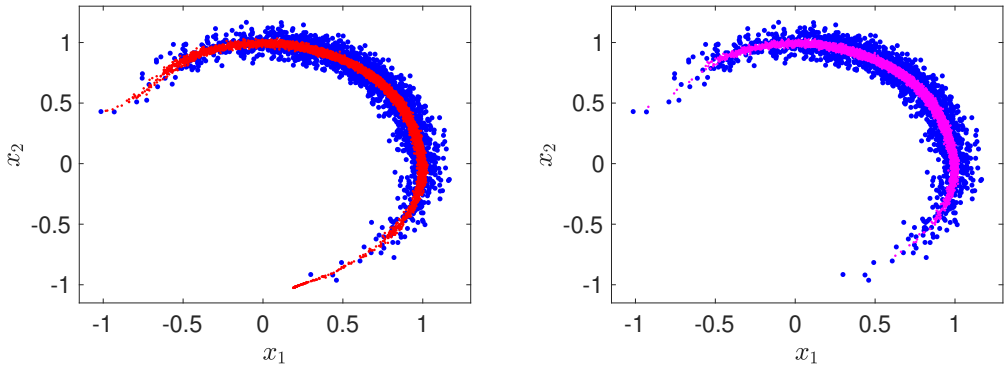

FIGURE 2. Effect of a variable bandwidth $K(x) = \rho(x)I$ in data-sparse regions. For the generative model the Langevin sampler (19) is used and we set $\epsilon = 0.009$. Results are shown for the output of step (19b). Left: Original (blue) and generated data for a constant bandwidth $K(x) = I$ (red). Right: Original (blue) and generated data for a variable bandwidth $K(x) = \rho(x)I$ with $\rho(x) = \pi(x)^\beta$ with $\beta = -1/5$ (magenta).

then setting $y^{(i)} = (z_1^{(i)}, \cdots, \alpha z_d^{(i)})$, with $\alpha = 5$ to promote non-uniformity. Finally, the samples $x^{(i)}$ are obtained by normalizing $y^{(i)}$ to achieve the unit length, i.e., $y^{(i)}/\left\|y^{(i)}\right\|$, and perturbing $y^{(i)}/\left\|y^{(i)}\right\|$ in the radial direction with $U(0, 0.01)$ noise. An instance of the target samples of three dimensions can be seen in Figure 5. Given a bandwidth ϵ and a bandwidth function $\rho(x)$, we implement the proposed scheme (19). For the fixed bandwidth kernel we set $\rho(x) = 1$. For the variable bandwidth kernel (22), we set $\rho(x) = \pi(x)^\beta$, with $\beta < 0$, and $\pi(x)$ is approximated using a kernel density estimator. We use $M = 1,000$ training samples to learn the transition kernel and run a Langevin sampler to generate $50,000$ samples, with the initial data point being $(1, 0, \cdots, 0) \in \mathbb{R}^d$. To obtain a better mixing of the Langevin sampler, we take one every 20 samples in the the last $2,0000$ generated samples of the chain, resulting in a total of $1,000$ samples. To evaluate the quality of the generated samples, we compute the regularized optimal transport (OT) distance between the generated samples $\{x_{\text{gen}}^{(i)}\}_{i=1}^{M_{\text{g}}}$ and the original reference samples $\{x_{\text{ref}}^{(j)}\}_{j=1}^{M_{\text{r}}}$. The regularized OT distance

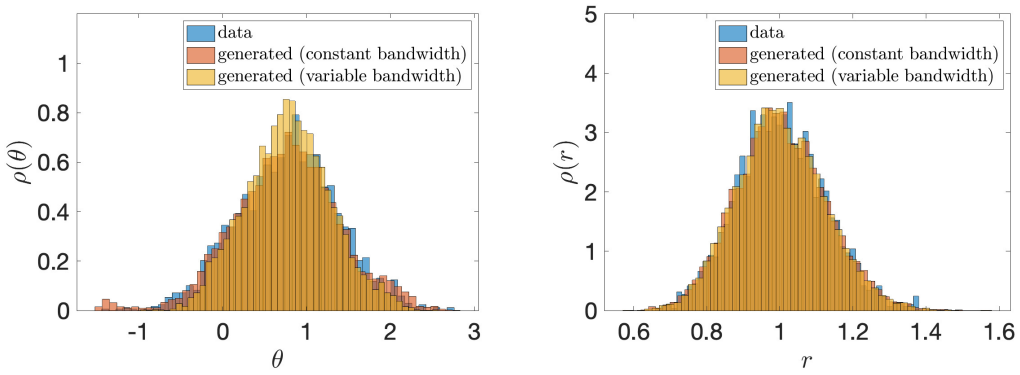

FIGURE 3. Effect of a variable bandwidth $K(x) = \rho(x)I$ on the angular and radial distributions (left and right, respectively). Shown are the original data (blue), generated data for a constant bandwidth $K(x) = I$ (red) and for a variable bandwidth $K(x) = \rho(x)I$ with $\rho(x) = \pi(x)^\beta$ with $\beta = -1/5$. The data were generated using a constant covariance noise model in (19) and $\epsilon = 0.009$.

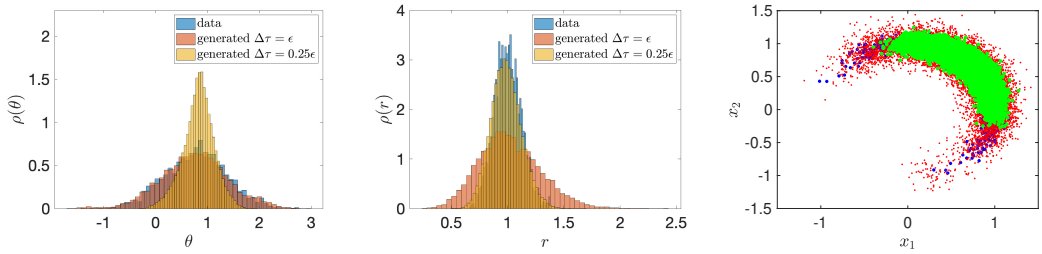

FIGURE 4. Effect of a variable time step $\Delta\tau$ in the Langevin sampler (18) with constant diffusion $K = 1$. Results are shown for the original data, and for $\Delta\tau = \epsilon$ and $\Delta\tau = \epsilon/4$. Throughout a constant bandwidth is used. Left: Empirical histogram of the angular variable θ. Middle: Empirical histograms of the radial variable r. Right: Original (blue) and generated data in the (x_1, x_2)-plane with $\Delta\tau = 1$ (red) and with $\Delta\tau = \epsilon/4$ (green).

with entropy penalty $1/\lambda$ is defined as

$$d_\lambda(x_{\text{gen}}, x_{\text{ref}}) = \min_P \sum_{i,j} P_{ij} C_{ij} - \frac{1}{\lambda} h(P),$$

subject to the constraint that

$$P 1_{M_{\text{r}}} = \frac{1}{M_{\text{g}}} (1, \cdots, 1)^\top \in \mathbb{R}^{M_g}$$
$$P^\top 1_{M_{\text{g}}} = \frac{1}{M_{\text{r}}} (1, \cdots, 1)^\top \in \mathbb{R}^{M_r},$$

where

$$h(P) = -\sum_{i,j} P_{ij} \log P_{ij}$$

d	optimal bandwidth
3	$\epsilon^* = 0.008$
4	$\epsilon^* = 0.010$
9	$\epsilon^* = 0.050$

TABLE 1. Optimal bandwidth parameters leading to minimal OT distance for different dimensions d, obtained using grid search.

is the information entropy. The entries of $C \in \mathbb{R}^{M_g \times M_r}$ are set to be the pairwise Euclidean distances between $\{x_{\text{gen}}^{(i)}\}_{i=1}^{M_{\text{g}}}$ and $\{x_{\text{ref}}^{(j)}\}_{j=1}^{M_{\text{r}}}$, and each sample is assigned equal weight marginally. We compute this distance using the Sinkhorn–Knopp algorithm [9, 26]. The number of reference samples is chosen to be $M_{\text{r}} = 50,000$, and the entropic regularization penalty is set to be $1/\lambda = 100$. We consider the OT distance as a diagnostic to quantify the statistical accuracy of the sampling scheme.

We then optimize over the parameters ϵ for a fixed λ using grid search. To be more precise, for the Langevin sampler with a fixed bandwidth kernel, we vary ϵ, and compute the OT distance of the generated samples. The best performed ϵ is chosen to be the one that corresponds to the smallest OT distance, and we call it ϵ^*. For the variable bandwidth kernel, we fix $\epsilon = \epsilon^*$. In order to disentangle the effect of varying ϵ and varying β we use the normalized variable bandwidth as described in (22). β is set to be -0.01×2^n with $n = \{0, \cdots, 8\}$ for $d = \{3, 4, 9\}$. The optimal bandwidth ϵ^* is reported in Table 1 and the comparisons between the fixed bandwidth kernel and the variable bandwidth kernel are presented in Figure 6. We observe that by keeping ϵ fixed, the OT distance becomes smaller for a wide range of β.

We then examine the quality of generated samples at the optimal ϵ and β along the last coordinate (the nonuniform direction). Similar to the previous study, we compute the one dimensional OT distance of the marginal distribution (see Figure 6b) and show the histograms and the cumulative density function (CDF) of the samples generated using the fixed bandwidth kernel and using the variable bandwidth kernel in Figure 7. The benefit of using the variable bandwidth kernel becomes more prominent when focusing on the marginal samples. In the case where we sample a 4-dimensional hyper-semisphere with non-uniformity along the last coordinate, we see in Figure 7 that the empirical CDF of the samples generated with the variable bandwidth kernel closely aligns with the reference (constructed using samples from the target distribution) for the most part. In contrast, the samples generated using the fixed variable bandwidth kernel noticeably diverge from the reference. This aligns with what we observed in Figure 2 and Figure 3 — the data generated using the variable bandwidth kernel better resemble the original data. In the cases where the samples are drawn from a 9-dimensional hyper-semisphere, while both methods struggle to generate samples that mirror those from the target distribution, primarily due to the inherent limitations of kernel methods in high-dimensional scenarios, employing a variable bandwidth kernel produces samples that exhibit a closer resemblance to those from the target distribution.

6.3. **Stochastic subgridscale parametrization.** The conditional sampling algorithm described in Section 5 can be used to perform stochastic subgridscale parametrization, a central problem encountered in, for example, the climate sciences. The problem of subgridscale parametrization, or more generally of model closure is the following:

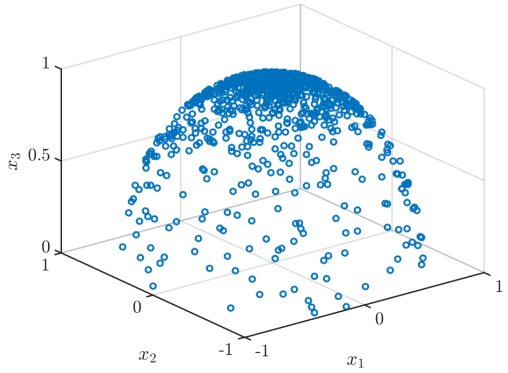

FIGURE 5. Nonuniform samples $x^{(i)}$ on a 3-dimensional semisphere. The non-uniformity is along the last coordinate x_3.

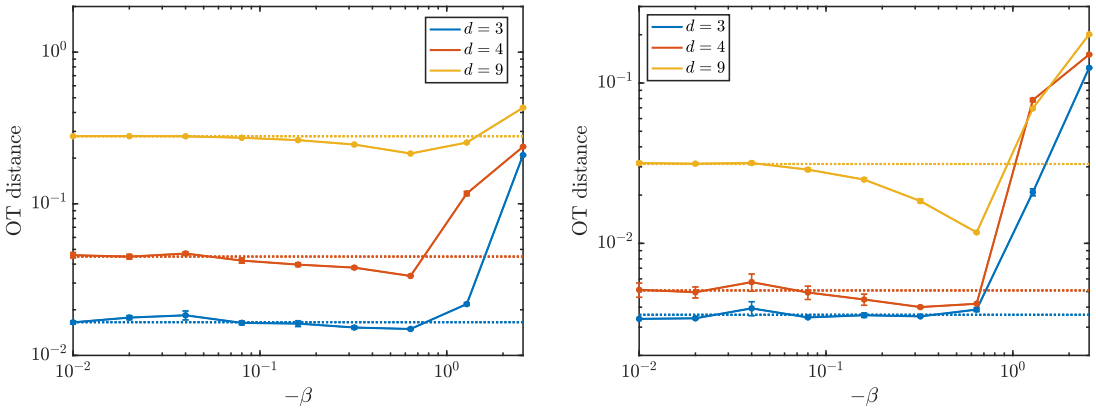

FIGURE 6. The dash lines are the OT distance of the samples using a fixed bandwidth kernel and the solid lines are the OT distance of the samples using a variable bandwidth kernel. (a) Comparison between the OT distance of samples generated using a fixed bandwidth kernel and using a variable bandwidth kernel. (b) Comparison between the *one-dimensional* marginal OT distance of samples generated using a fixed bandwidth kernel and using a variable bandwidth kernel, along the last coordinate (non-uniform direction). Here the Langevin sampler is initialized at $(1, 0, \cdots, 0)$ for all cases.

given a potentially stiff dynamical system

$$\dot{z} = F_z(z, y) + g(z, y; \varepsilon) \tag{26a}$$

$$\dot{y} = F_y(z, y; \varepsilon), \tag{26b}$$

where $\varepsilon < 1$ denotes the time scale separation between slow resolved variables of interest $z \in \mathbb{R}^{d_s}$ and unresolved fast degrees $y \in \mathbb{R}^{d_f}$. Note the notational difference between the time scale separation parameter ε and the bandwidth parameter ϵ used to define the diffusion map. For $\varepsilon \ll 1$ the system is stiff and to ensure numerical stability a small time step $\Delta t < \varepsilon$ is needed. This, together with the potential high-dimensionality of the fast subspace $d_f \gg d_s$, constitutes a computational barrier for

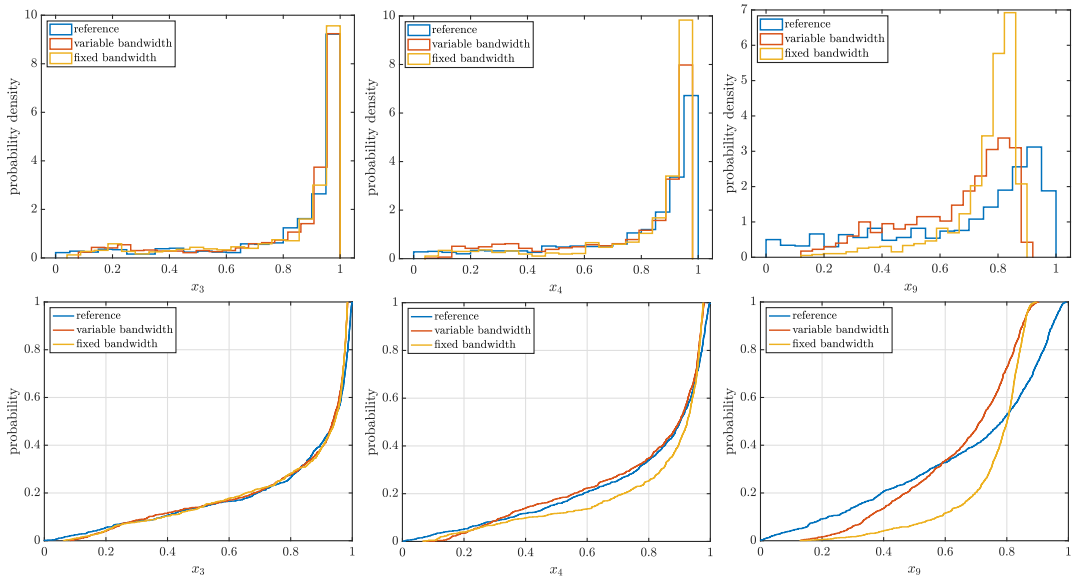

FIGURE 7. Comparisons of empirical histograms (top row) and CDFs (bottom row) of the marginal distribution of the generated samples along the last coordinate. From left to right: the data are sampled from a $\{3, 4, 9\}$-dimensional (hyper-)semisphere.

simulating the dynamics on the slow time scale of interest. Hence one is interested in obtaining an effective evolution equation for the slow resolved variables z only which captures the essential effect of the unresolved variables y. Hence, we seek to determine the effective reduced dynamics

$$\dot{z} = F_z(z) + \psi(z), \tag{27}$$

where $F_z(z)$ denotes some *a priori* known deterministic drift, possibly based on physical reasoning, and $\psi(z)$ denotes the closure term which may be deterministic or stochastic, and which parametrizes the unknown unresolved fast processes. Deterministic machine learning methods have previously been used to learn the average effect of the unresolved variables, i.e. the average of $g(z, y; \varepsilon)$ over the (conditional) invariant measure of the fast process [18, 19]. Deterministic maps, however, are not able to capture the resolved dynamics with sufficient statistical accuracy, and it is by now well established that the effective equation is of a stochastic nature [34, 17, 23, 16]. In the climate sciences this idea goes back to the seminal work by the 2022 Nobel Prize recipient Klaus Hasselmann who treated the slow ocean as a stochastic dynamical system as a result of it experiencing the integrated effect of fast moving weather systems. In the case of infinite time scale separation there are explicit expressions for the effective drift and diffusion term of the effective slow dynamics. However, these terms include integrals over the auto-correlation functions and are numerically very hard to estimate. Moreover, for the realistic case of moderate time-scale separation Edgeworth corrections need to be included [59, 60], the estimation of which requires even longer time-series. Instead we propose to learn the closure term $\psi(z)$ and generate realisations $\psi(z)$ on the fly employing the conditional sampling algorithm described above. We consider the situation in which scientists have a good understanding of the resolved dynamics and know the slow vector field $F_z(z)$. Given data of the resolved variables $\{z_n\}_{n=1}^N$ sampled at equidistant times Δt, the closure term $\psi(z)$ capturing the effect of the

unresolved dynamics (26b) can then be estimated as

$$\psi(z_n) = z_{n+1} - z_n - F_z(z_n)\,\Delta t, \tag{28}$$

for $n = 1, \cdots, M-1$. Note that the closure term $\psi(z)$ typically includes effective diffusion as well as a correction to the drift term $F_z(z)$ [15]. The effective dynamics is then provided by the discrete stochastic surrogate model

$$z_{n+1} = z_n + F_z(z_n)\,\Delta t + \psi_n, \tag{29}$$

where the subgrid-scale terms $\psi_n = \psi(z_n)$ are generated as follows: Given a kernel $t_i(x)$ with $x = (z, \psi(z))^T$ and a v_ϵ obtained by the Sinkhorn algorithm as described in Section 3, and the $2d_s \times M$ data matrix $\mathcal{X}$ consisting of samples $\{x^{(j)}\}_{j=1}^M$, we perform for $j = 1, \cdots, n_s$ with $n_s = 100$ a discrete Langevin sampler for x

$$\begin{aligned} z^* &= z_n \\ x_{1,j} &= z^* \\ x_{j+1/2} &= x_j + \sqrt{2\epsilon}\,\Xi_j \\ x_{j+1} &= \mathcal{Z} p_\epsilon(x_{j+1/2}), \end{aligned}$$

with $\Xi_n \sim \mathrm{N}(0, I)$. The first assignment $x_{1,n} = z^* = z_n$ ensures the conditioning of $\psi(z_n)$ on $z^* = z_n$, and $n_s = 100$ ensures that the generated samples $\psi(z_n)$ are close to independent. We choose a fixed bandwidth with $\epsilon = 0.001$.

We consider here the particular example with $d_s = 1$ and $d_f = 3$ given by

$$\dot{z} = z(1 - z^2) + \frac{4}{90\varepsilon} h(z)\, y_2, \tag{30}$$

where the fast dynamics is given by the Lorenz-63 system [31]

$$\varepsilon^2 \dot{y}_1 = 10(y_2 - y_1) \tag{31a}$$

$$\varepsilon^2 \dot{y}_2 = 28 y_1 - y_2 - y_1 y_3 \tag{31b}$$

$$\varepsilon^2 \dot{y}_3 = -\frac{8}{3} y_3 + y_1 y_2. \tag{31c}$$

We look at the case of effective additive noise with $h(z) = 1$ which does not require conditioning on the slow variable z, as well as at the case of multiplicative noise with $h(z) = z$. We used MATLAB's built-in ode45 routine [32] to generate the time series with a time-scale separation parameter of $\varepsilon = 0.01$. The time series is subsequently sub-sampled with $\Delta t = 0.1$. Figure 8 shows a comparison between the full multi-scale system (30) - (31) with additive noise $h(z) = 1$ and the stochastic surrogate model (29). The slow z-dynamics exhibits stochastic bimodal dynamics. Figure 9 shows a comparison for the multiplicative case $h(z) = z$ which yields unimodal slow dynamics. It is clearly seen that the surrogate model (29) obtained by the generative conditional sampler generates statistically reliable dynamics.

7. Conclusions

We introduced a diffusion map based Langevin scheme for generative modeling. Our method combines diffusion maps with discrete-time Langevin-type sampling, yielding a nonparametric generative model that requires minimal tuning. We showed numerically that employing a variable bandwidth kernel, in contrast to a fixed bandwidth kernel, results in generated samples with enhanced accuracy. The performance of the conditional generative model was showcased through its application to a stochastic subgrid-scale parametrization problem. Future research will delve into the theoretical foundations of the proposed scheme, including its convergence rate and scalability.

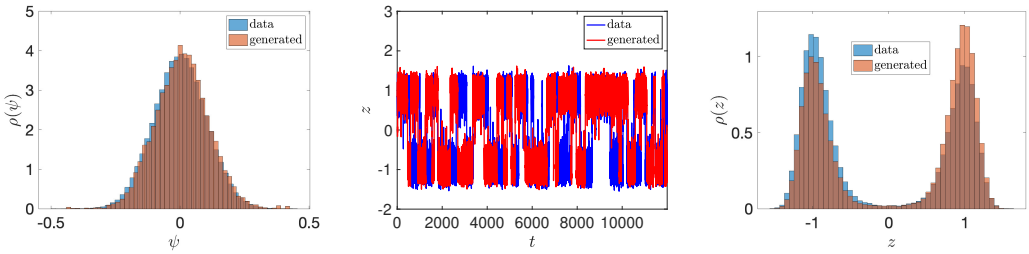

FIGURE 8. Results for the stochastic subgrid-scale parametrization for the multi-scale system (30) - (31) with $\varepsilon = 0.01$ and with additive noise $h(z) = 1$. Shown are results obtained by integrating the full multi-scale system and by the stochastic subgridscale parametrization scheme using our generative sampler trained with $M = 120,000$. Left: Empirical histograms of the closure term $\psi = \psi(z)$. Middle: Time series of the slow variable $z(t)$. Right: Empirical histograms of the slow variable z.

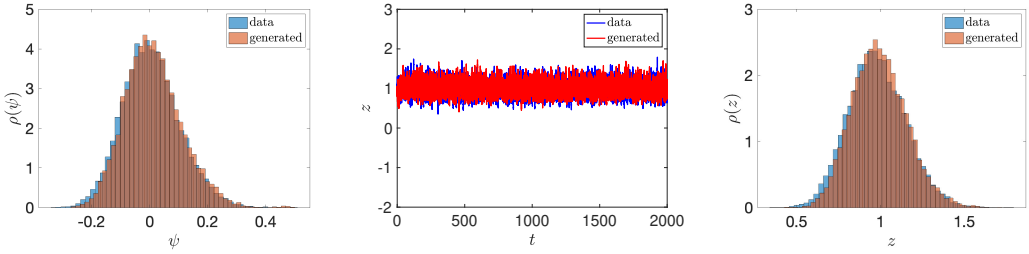

FIGURE 9. Results for the stochastic subgrid-scale parametrization for the multi-scale system (30) - (31) with $\varepsilon = 0.01$ and with multiplicative noise $h(z) = z$. Shown are results obtained by integrating the full multi-scale system and by the stochastic subgridscale parametrization scheme using our generative sampler, trained with $M = 20,000$. Left: Empirical histograms of the closure term $\psi = \psi(z)$. Middle: Time series of the slow variable $z(t)$. Right: Empirical histograms of the slow variable z.

Acknowledgements. This work has been funded by Deutsche Forschungsgemeinschaft (DFG) - Project-ID 318763901 - SFB1294. GAG acknowledges funding from the Australian Research Council, grant DP220100931. FL and YMM acknowledge support from the US Department of Energy, Office of Advanced Scientific Computing Research, award DE-SC0023188. We thank Ricardo Baptista for pointing out the relationship to the Barycentric projection.

REFERENCES

[1] T. Berry and J. Harlim. Variable bandwidth diffusion kernels. *Applied and Computational Harmonic Analysis*, 40(1):68–96, 2016. ISSN 1063-5203. doi: https://doi.org/10.1016/j.acha.2015.01.001. URL `https://www.sciencedirect.com/science/article/pii/S1063520315000020`.

[2] Y. Bian and X.-Q. Xie. Generative chemistry: drug discovery with deep learning generative models. *Journal of Molecular Modeling*, 27:1–18, 2021.

[3] C. Bilodeau, W. Jin, T. Jaakkola, R. Barzilay, and K. F. Jensen. Generative models for molecular discovery: Recent advances and challenges. *WIREs*

Computational Molecular Science, 12(5):e1608, 2022. doi: https://doi.org/10.1002/wcms.1608. URL https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wcms.1608.

[4] A. Block, Y. Mroueh, and A. Rakhlin. Generative modeling with denoising auto-encoders and langevin sampling. *ArXiv*, abs/2002.00107, 2020. URL https://api.semanticscholar.org/CorpusID:211010797.

[5] R. Cai, G. Yang, H. Averbuch-Elor, Z. Hao, S. J. Belongie, N. Snavely, and B. Hariharan. Learning gradient fields for shape generation. In A. Vedaldi, H. Bischof, T. Brox, and J. Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part III*, volume 12348 of *Lecture Notes in Computer Science*, pages 364–381. Springer, 2020. doi: 10.1007/978-3-030-58580-8_22. URL https://doi.org/10.1007/978-3-030-58580-8_22.

[6] J. A. Carrillo and U. Vaes. Wasserstein stability estimates for covariance-preconditioned Fokker–Planck equations. *Nonlinearity*, 34(4):2275, feb 2021. doi: 10.1088/1361-6544/abbe62. URL https://dx.doi.org/10.1088/1361-6544/abbe62.

[7] R. R. Coifman and S. Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, 2006. ISSN 1063-5203. doi: https://doi.org/10.1016/j.acha.2006.04.006. URL https://www.sciencedirect.com/science/article/pii/S1063520306000546. Special Issue: Diffusion Maps and Wavelets.

[8] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker. Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. *Proceedings of the National Academy of Sciences*, 102(21):7426–7431, 2005. doi: 10.1073/pnas.0500334102. URL https://www.pnas.org/doi/abs/10.1073/pnas.0500334102.

[9] M. Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL https://proceedings.neurips.cc/paper/2013/file/af21d0c97db2e27e13572cbf59eb343d-Paper.pdf.

[10] L. Deecke, R. Vandermeulen, L. Ruff, S. Mandt, and M. Kloft. Image anomaly detection with generative adversarial networks. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2018, Dublin, Ireland, September 10–14, 2018, Proceedings, Part I 18*, pages 3–17. Springer, 2019.

[11] J. Fan and I. Gijbels. Variable bandwidth and local linear regression smoothers. *The Annals of Statistics*, 20(4):2008 – 2036, 1992. doi: 10.1214/aos/1176348900. URL https://doi.org/10.1214/aos/1176348900.

[12] A. Garbuno-Inigo, F. Hoffmann, W. Li, and A. M. Stuart. Interacting Langevin diffusions: Gradient structure and ensemble Kalman sampler. *SIAM Journal on Applied Dynamical Systems*, 19(1):412–441, 2020. doi: 10.1137/19M1251655. URL https://doi.org/10.1137/19M1251655.

[13] M. Girolami and B. Calderhead. Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(2):123–214, 2011. doi: https://doi.org/10.1111/j.1467-9868.2010.00765.x. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2010.00765.x.

[14] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf.

[15] G. Gottwald and I. Melbourne. Time-reversibility and nonvanishing Lévy area, 2022.

[16] G. Gottwald, D. Crommelin, and C. Franzke. Stochastic climate theory. In C. L. E. Franzke and T. J. O'Kane, editors, *Nonlinear and Stochastic Climate Dynamics*, pages 209–240. Cambridge University Press, Cambridge, 2017.

[17] G. A. Gottwald and I. Melbourne. Homogenization for deterministic maps and multiplicative noise. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 469(2156), 2013.

[18] G. A. Gottwald and S. Reich. Supervised learning from noisy observations: Combining machine-learning techniques with data assimilation. *Physica D: Nonlinear Phenomena*, 423:132911, 2021. ISSN 0167-2789. doi: https://doi.org/10.1016/j.physd.2021.132911. URL https://www.sciencedirect.com/science/article/pii/S0167278921000695.

[19] G. A. Gottwald and S. Reich. Combining machine learning and data assimilation to forecast dynamical systems from noisy partial observations. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 31(10):101103, 2021. ISSN 1054-1500. doi: 10.1063/5.0066080. URL https://doi.org/10.1063/5.0066080.

[20] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

[21] J. Ho, C. Saharia, W. Chan, D. J. Fleet, M. Norouzi, and T. Salimans. Cascaded diffusion models for high fidelity image generation. *J. Mach. Learn. Res.*, 23:47:1–47:33, 2021.

[22] A. Hyvärinen and P. Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005.

[23] D. Kelly and I. Melbourne. Deterministic homogenization for fast–slow systems with chaotic noise. *Journal of Functional Analysis*, 272(10):4063 – 4102, 2017.

[24] A. A. S. Khandelwal. *Fine-tuning generative models*. PhD thesis, Massachusetts Institute of Technology, 2019.

[25] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *CoRR*, abs/1312.6114, 2013.

[26] P. A. Knight. The Sinkhorn-Knopp algorithm: Convergence and applications. *SIAM J. Matrix Anal. Appl.*, 30:261–275, 2008.

[27] R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. Bayesian imaging using plug & play priors: When Langevin meets Tweedie. *SIAM Journal on Imaging Sciences*, 15(2):701–737, 2022. doi: 10.1137/21M1406349. URL https://doi.org/10.1137/21M1406349.

[28] C. Li, C. Chen, D. E. Carlson, and L. Carin. Preconditioned stochastic gradient langevin dynamics for deep neural networks. In *AAAI Conference on Artificial Intelligence*, 2015. URL https://api.semanticscholar.org/CorpusID:17043130.

[29] X. L. Li, J. Thickstun, I. Gulrajani, P. Liang, and T. B. Hashimoto. Diffusion-lm improves controllable text generation, 2022. URL https://arxiv.org/abs/2205.14217.

[30] Y. Li, Q. Pan, S. Wang, T. Yang, and E. Cambria. A generative model for category text generation. *Information Sciences*, 450:301–315, 2018. ISSN 0020-0255. doi: https://doi.org/10.1016/j.ins.2018.03.050. URL https://www.sciencedirect.com/science/article/pii/S0020025518302366.

[31] E. N. Lorenz. Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, 20(2):130–141, 1963. doi: 10.1175/1520-0469(1963)020⟨0130:DNF⟩2.0.CO;2.

[32] MATLAB. *version 9.13.0.2049777 (R2022b)*. The MathWorks Inc., Natick, Massachusetts, 2022.

[33] J. Mattingly, A. Stuart, and D. Higham. Ergodicity for SDEs and approximations: locally Lipschitz vector fields and degenerate noise. *Stochastic Processes and their Applications*, 101:185–232, 2002.

[34] I. Melbourne and A. Stuart. A note on diffusion limits of chaotic skew-product flows. *Nonlinearity*, 24:1361–1367, 2011.

[35] S. Meyn and R. T. Tweedy. *Markov Chains and Stochastic Stability*. Cambridge University Press, Cambridge, 2nd edition, 2009. doi: 10.1017/CBO9780511626630.

[36] H.-G. Muller and U. Stadtmuller. Variable bandwidth kernel estimators of regression curves. *The Annals of Statistics*, 15(1):182 – 201, 1987. doi: 10.1214/aos/1176350260. URL https://doi.org/10.1214/aos/1176350260.

[37] B. Nadler, S. Lafon, I. Kevrekidis, and R. Coifman. Diffusion maps, spectral clustering and eigenfunctions of Fokker-Planck operators. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems*, volume 18. MIT Press, 2005. URL https://proceedings.neurips.cc/paper/2005/file/2a0f97f81755e2878b264adf39cba68e-Paper.pdf.

[38] B. Nadler, S. Lafon, R. R. Coifman, and I. G. Kevrekidis. Diffusion maps, spectral clustering and reaction coordinates of dynamical systems. *Applied and Computational Harmonic Analysis*, 21(1):113–127, 2006. ISSN 1063-5203. doi: https://doi.org/10.1016/j.acha.2005.07.004. URL https://www.sciencedirect.com/science/article/pii/S1063520306000534. Special Issue: Diffusion Maps and Wavelets.

[39] N. Nüsken and S. Reich. Note on interacting Langevin diffusions: Gradient structure and ensemble Kalman sampler by Garbuno-Inigo, Hoffmann, Li and Stuart, 2019.

[40] A.-A. Pooladian and J. Niles-Weed. Entropic estimation of optimal transport maps, 2022. URL https://arxiv.org/pdf/2109.12004.pdf.

[41] L. Rey-Bellet and K. V. Spiliopoulos. Improving the convergence of reversible samplers. *Journal of Statistical Physics*, 164:472–494, 2016.

[42] D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pages 1278–1286. PMLR, 2014.

[43] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models, 2021. URL https://arxiv.org/abs/2112.10752.

[44] L. Ruthotto and E. Haber. An introduction to deep generative modeling. *GAMM-Mitteilungen*, 44(2):e202100008, 2021. doi: https://doi.org/10.1002/gamm.202100008. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/gamm.

202100008.

[45] I. H. Salgado-Ugarte and M. A. Pérez-Hernández. Exploring the use of variable bandwidth kernel density estimators. *The Stata Journal*, 3(2):133–147, 2003. doi: 10.1177/1536867X0300300203. URL https://doi.org/10.1177/1536867X0300300203.

[46] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen. Improved techniques for training GANs. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/8a3363abe792db2d8761d6403605aeb7-Paper.pdf.

[47] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International conference on information processing in medical imaging*, pages 146–157. Springer, 2017.

[48] V. Seguy, B. B. Damodaran, R. Flamary, N. Courty, A. Rolet, and M. Blondel. Large-scale optimal transport and mapping estimation. In *Proceedings of the International Conference in Learning Representations*, 2018.

[49] T. R. Shaham, T. Dekel, and T. Michaeli. Singan: Learning a generative model from a single natural image. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4570–4580, 2019.

[50] J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[51] Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/3001ef257407d5a371a96dcd947c7d93-Paper.pdf.

[52] Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[53] Y. Song and S. Ermon. Improved techniques for training score-based generative models. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 12438–12448. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/92c3b916311a5517d9290576e3ea37ad-Paper.pdf.

[54] Y. Song, S. Garg, J. Shi, and S. Ermon. Sliced score matching: A scalable approach to density and score estimation. In *Uncertainty in Artificial Intelligence*, pages 574–584. PMLR, 2020.

[55] Y. Song, J. N. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. *ArXiv*, abs/2011.13456, 2020.

[56] G. R. Terrell and D. W. Scott. Variable kernel density estimation. *The Annals of Statistics*, pages 1236–1265, 1992.

[57] Z. Wang, H. Zheng, P. He, W. Chen, and M. Zhou. Diffusion-gan: Training GANs with diffusion. *arXiv preprint arXiv:2206.02262*, 2022.

[58] C. Wormell and S. Reich. Spectral convergence of diffusion maps: Improved error bounds and an alternative normalisation. *SIAM J. Numer. Anal.*, 59:1687–1734, 2021. doi: 10.1137/20M1344093.

[59] J. Wouters and G. A. Gottwald. Edgeworth expansions for slow–fast systems with finite time-scale separation. *Proceedings of the Royal Society A: Mathematical,*

Physical and Engineering Sciences, 475(2223):20180358, 2019.

[60] J. Wouters and G. A. Gottwald. Stochastic model reduction for slow-fast systems with moderate time scale separation. *Multiscale Modeling & Simulation*, 17(4): 1172–1188, 2019.

[61] T. Xifara, C. Sherlock, S. Livingstone, S. Byrne, and M. Girolami. Langevin diffusions and the Metropolis-adjusted Langevin algorithm. *Statistics & Probability Letters*, 91(C):14–19, 2014. doi: 10.1016/j.spl.2014.04.002. URL `https://ideas.repec.org/a/eee/stapro/v91y2014icp14-19.html`.

[62] L. Yang, Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, Y. Shao, W. Zhang, B. Cui, and M.-H. Yang. Diffusion models: A comprehensive survey of methods and applications, 2022. URL `https://arxiv.org/abs/2209.00796`.

[63] D. Yogatama, C. Dyer, W. Ling, and P. Blunsom. Generative and discriminative text classification with recurrent neural networks. *ArXiv*, abs/1703.01898, 2017.