Diffusion Generative Models in Infinite Dimensions

Gavin Kerrigan
University of California, Irvine
gavin.k@uci.edu

Justin Ley
University of California, Irvine
jsley@uci.edu

Padhraic Smyth
University of California, Irvine
smyth@ics.uci.edu

Abstract

Diffusion generative models have recently been applied to domains where the available data can be seen as a discretization of an underlying function, such as audio signals or time series. However, these models operate directly on the discretized data, and there are no semantics in the modeling process that relate the observed data to the underlying functional forms. We generalize diffusion models to operate directly in function space by developing the foundational theory for such models in terms of Gaussian measures on Hilbert spaces. A significant benefit of our function space point of view is that it allows us to explicitly specify the space of functions we are working in, leading us to develop methods for diffusion generative modeling in Sobolev spaces. Our approach allows us to perform both unconditional and conditional generation of function-valued data. We demonstrate our methods on several synthetic and real-world benchmarks.

1 Introduction

Diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020) have recently emerged as a powerful class of generative models on a wide array of domains, ranging from images (Ho et al., 2020; Dhariwal and Nichol, 2021; Saharia et al., 2022; Ramesh et al., 2022) and video (Ho et al., 2022; Yang et al., 2022) to molecular conformation (Xu et al., 2022). At an intuitive level, these methods work by iteratively perturbing the data distribution towards a tractable prior via additive Gaussian noise, and generation is performed by learning to undo this transformation.

Existing methods largely assume that the data distribution of interest is supported on a finite-dimensional Euclidean space. However, in many domains, the underlying signal is inherently *infinite-dimensional*, where the observed data can be seen as a collection of discrete observations of some underlying function. Such datasets are often dubbed *functional* (Ramsay and Silverman, 2008). For instance, a time series dataset consisting of the temperature collected at a particular location every 24 hours can be seen as a uniform discretization of an underlying continuous-time temperature curve (Febrero-Bande and de la Fuente, 2012).

Although diffusion models have empirically demonstrated strong performance on some functional domains, such as audio signals (Kong et al., 2021; Chen et al., 2021) and time series (Rasul et al., 2021; Tashiro et al., 2021; Alcaraz and Strodthoff, 2022), existing approaches work directly on an explicit discretization of the input space. It is thus unclear how existing methods relate to the underlying functions of interest. For instance, existing methods can not account for function-level assumptions about the data, such as continuity or smoothness constraints.

Motivated by a functional perspective, we propose a novel theoretical framework for diffusion generative modeling which operates directly in function space. Our primary contributions are as follows:

- In Section 4, we develop a framework for diffusion generative modeling in terms of Gaussian measures on Hilbert spaces. Our method operates by adding Gaussian process noise directly to our infinite-dimensional functions. We learn to reverse this process by performing variational inference in function space, in which we minimize the KL divergence between a known Gaussian measure and a variational family of Gaussian measures. We discuss the necessary background on Gaussian measures in Section 3.
- In Section 5, we propose practical methods for approximating functional KL divergences by discretizing the underlying operators. The practical details depend heavily on the choice of function space, and we develop methods for the space of square-integrable functions as well as Sobolev spaces.
- In Section 6, we empirically verify our framework on several synthetic and real-world benchmarks. In our

Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206.

experiments, our diffusion models are implemented via neural networks that parametrize mappings between function spaces, i.e. neural operators (Li et al., 2021, 2020; Kovachki et al., 2021). We propose methods that allow for both unconditional and conditional generation of function-valued data. Importantly, our approach allows us to work with arbitrary non-uniform discretizations, thereby allowing us to train on datasets where the observation set varies across functions. Moreover, we are able to query our generated functions at arbitrary input locations.

2 Related Work

Diffusion models are most typically applied to data living in a Euclidean space having a fixed, finite dimension (e.g., see Sohl-Dickstein et al. (2015); Ho et al. (2020); Dhariwal and Nichol (2021); Ho et al. (2020) amongst others). More recent work has extended these methods to Riemannian manifolds, but still with a finite-dimensional assumption (Bortoli et al., 2022; Huang et al., 2022).

Most relevant to our work are diffusion models for signals, such as audio (Chen et al., 2021; Kong et al., 2021), time series (Rasul et al., 2021; Tashiro et al., 2021; Alcaraz and Strodthoff, 2022), or neural processes (Dutordoir et al., 2022). However, these current approaches for functional data all perform diffusion modeling by employing standard finite-dimensional diffusion modeling on the discretized functions. Concurrent to our work, Biloš et al. (2022) propose a diffusion model for temporal data, but do not take a function space perspective. As we will show in Section 5.3, existing approaches can be viewed as special cases within the general theoretical framework we develop.

Subsequent to our work in this paper, Lim et al. (2023) and Pidstrigach et al. (2023) in follow-up work proposed methodologies which are closely related and conceptually similar to our approach. As in our work, Lim et al. (2023) and Pidstrigach et al. (2023) both perturbed the function space distribution corresponding to the data via a trace-class Gaussian measure. Our work can be seen as extending the discrete time DDPM model (Ho et al., 2020) to function spaces, while the works of Lim et al. (2023) and Pidstrigach et al. (2023) can be seen as extending score-matching techniques (Vincent, 2011; Song and Ermon, 2019) to function spaces. In particular, Lim et al. (2023) developed techniques for function space score matching in discrete time, and Pidstrigach et al. (2023) developed function space score matching techniques from a continuous time perspective.

Beyond diffusion models, a recent line of work has proposed deep generative models of functions (Garnelo et al., 2018; Kim et al., 2019; Dupont et al., 2022b,a). In particular, generative models of functions based on neural operators have been proposed from a GAN approach (Rahman et al., 2022). However, ours is the first work to combine diffusion models with neural operators.

Our approach is also broadly related to the general class of previous works that propose function-space perspectives in machine learning. In particular, such a point of view has proved useful for developing and understanding techniques used in Gaussian processes (Matthews et al., 2016; Wynne and Wild, 2022) and Bayesian deep learning (Sun et al., 2019; Wild et al., 2022; Rudner et al., 2021; Tran et al., 2022; Burt et al., 2021). Our work extends this function-space perspective to diffusion models.

3 Notation and Background

We begin by setting up the notation for our problem and introducing the necessary background on Gaussian measures in Hilbert spaces, as well as their connection to the more familiar notion of Gaussian processes. In addition, we derive a closed-form expression for the KL divergence between Gaussian measures with equal covariance operators – this KL divergence plays a key role in our approach.

3.1 Notation and Data

Let $(\mathcal{X}, \mathcal{A}, \mu)$ be a measure space where $\mathcal{X} \subseteq \mathbb{R}^{d_x}$, and let $\mathcal{F}$ be a separable Hilbert space of measurable real-valued functions on $\mathcal{X}$ with inner product $\langle -, - \rangle_{\mathcal{F}}$. Note that we will often simply write $\langle -, - \rangle$ if the choice of $\mathcal{F}$ is clear from context. We equip $\mathcal{F}$ with its Borel σ-algebra $\mathcal{B}(\mathcal{F})$. The prototypical example is $\mathcal{X} = [0, 1]$ equipped with the Lebesgue measure and $\mathcal{F} = L^2(\mathcal{X}, \mu)$ equipped with its usual inner product $\langle f, g \rangle_{L^2(\mathcal{X},\mu)} = \int_{\mathcal{X}} fg \, \mathrm{d}\mu$. However, our general framework is agnostic to the choice of $\mathcal{F}$ – see Section 5 for more details on this choice.

We assume that we have a dataset of the form $\mathcal{D} = \{u^{(1)}, u^{(2)}, \ldots, u^{(n)}\}$, where each $u^{(j)} \in \mathcal{F}$ is an i.i.d. draw from an unknown probability measure $\mathbb{P}_{\text{data}}$ on $\mathcal{F}$. In practice, we typically only have noisy measurements of our functions on a finite subset of $\mathcal{X}$. We let $\vec{x}^{(j)} = \{x^{(1j)}, \ldots, x^{(mj)}\} \subset \mathcal{X}$ be a discrete subset of $\mathcal{X}$ with corresponding observations $\vec{y}^{(j)} = \{y^{(1j)}, \ldots, y^{(mj)}\}$, where $y^{(ij)} = u^{(j)}(x^{(ij)}) + \epsilon^{(ij)}$ is the output of the unknown jth function $u^{(j)}$ at the ith observation point and $\epsilon^{(ij)}$ represents i.i.d. observation noise. Generally, both the location $\vec{x}^{(j)}$ and number $m = m_j$ of observation points may vary across the functions in our dataset.

The focus of this work is to develop the theory and practice behind building a diffusion generative model for sampling from the function-space probability measure $\mathbb{P}_{\text{data}}$.

3.2 Gaussian Measures

We now introduce some key background material on Gaussian measures (Da Prato and Zabczyk, 2014).

Definition 1.
Let $(\Omega, \mathcal{B}, \mathbb{P})$ be a probability space. A measurable function $F : \Omega \to \mathcal{F}$ is called a *Gaussian random element (GRE)* if for any $g \in \mathcal{F}$, the random variable $\langle g, F \rangle$ has a Gaussian distribution on $\mathbb{R}$. The pushforward of $\mathbb{P}$ along F, denoted $\mathbb{P}_F = F_{\#}\mathbb{P}$ is a *Gaussian (probability) measure* on $\mathcal{F}$.

Gaussian random elements $F \sim \mathbb{P}_F$ are random functions in $\mathcal{F}$. Note that Gaussian measures exactly coincide with the standard notion of Gaussian distributions in the special case of $\mathcal{F} = \mathbb{R}^n$ equipped with the usual inner product.

For every GRE $F \sim \mathbb{P}_F$, there exists a unique mean element $m \in \mathcal{F}$ given by

$$m = \int_{\mathcal{F}} F \, \mathrm{d}\mathbb{P}_F. \tag{3.1}$$

Similarly, there exists a unique linear covariance operator $C : \mathcal{F} \to \mathcal{F}$ given by

$$Cg = \int_{\mathcal{F}} \langle g, F \rangle F \, \mathrm{d}\mathbb{P}_F - \langle g, m \rangle m \quad \forall g \in \mathcal{F}. \tag{3.2}$$

A Gaussian measure is uniquely determined by its mean element and covariance operator. Note that for any $g \in \mathcal{F}$, we have that $\langle g, F \rangle \sim \mathcal{N}(\langle g, m \rangle, \langle Cg, g \rangle)$ follows a Gaussian distribution on $\mathbb{R}$ with mean $\langle g, m \rangle \in \mathbb{R}$ and variance $\sigma^2 = \langle Cg, g \rangle \in \mathbb{R}_{\geq 0}$ (Wild et al., 2022).

The covariance operator C is symmetric, positive semidefinite, and compact. Moreover, C has finite trace, i.e. $\mathrm{tr}(C) = \mathbb{E}[||F||^2] < \infty$. Conversely, given any $m' \in \mathcal{F}$ and any symmetric, positive semidefinite, trace-class linear operator $C' : \mathcal{F} \to \mathcal{F}$, there exists a Gaussian measure having mean m' and covariance operator C'. Thus, Gaussian measures are in one-to-one correspondence with their mean functions and covariance operators. We will write $\mathbb{P}_F = \mathcal{N}(m, C)$ for such a Gaussian measure. We refer to Da Prato and Zabczyk (2014, Chapter 2) and Bogachev (1998) for the proofs of these claims.

3.3 KL Divergence between Gaussian Measures

In our framework, we will perform variational inference in function space. However, one major challenge is that there is no analogue of the Lebesgue measure on infinite dimensional spaces (Eldredge, 2016), and so we must resort to a measure-theoretic definition of the KL divergence. To that end, for probability measures $\mathbb{P}, \mathbb{Q}$ on $\mathcal{F}$, we define

$$\mathsf{KL}[\mathbb{P} \parallel \mathbb{Q}] = \int_{\mathcal{F}} \log \left(\frac{\mathrm{d}\mathbb{P}}{\mathrm{d}\mathbb{Q}} \right) \mathrm{d}\mathbb{P} \tag{3.3}$$

if $\mathbb{P} \ll \mathbb{Q}$, where $\mathrm{d}\mathbb{P}/\,\mathrm{d}\mathbb{Q}$ is the Radon-Nikodym derivative. We define this quantity to be infinite if $\mathbb{P}$ is not absolutely continuous with respect to $\mathbb{Q}$.

We now consider the special case that $\mathbb{P}, \mathbb{Q}$ are Gaussian measures on $\mathcal{F}$ with equal covariance operators. In this case, a version of the Feldman-Hájek Theorem gives us explicit control over the Radon-Nikodym derivative in terms of the parameters of $\mathbb{P}$ and $\mathbb{Q}$ (Da Prato and Zabczyk, 2014, Theorem 2.23).

Theorem 1 (The Feldman-Hájek Theorem)**.**
Let $\mathbb{P} = \mathcal{N}(m_1, C)$ and $\mathbb{Q} = \mathcal{N}(m_2, C)$ be Gaussian measures on $\mathcal{F}$ with equal covariance operators, and define $\Delta m = m_1 - m_2 \in \mathcal{F}$. Then, $\mathbb{P}$ and $\mathbb{Q}$ are equivalent (i.e. mutually absolutely continuous) if and only if $\Delta m \in C^{1/2}(\mathcal{F})$. In this case, for any $f \in \mathcal{F}$, the Radon-Nikodym derivative $\mathrm{d}\mathbb{P}/\,\mathrm{d}\mathbb{Q}$ is given by

$$\frac{\mathrm{d}\mathbb{P}}{\mathrm{d}\mathbb{Q}}(f) = \exp\big[\left\langle \Delta m, C^{-1}(f - m_2) \right\rangle_{\mathcal{F}} - \frac{1}{2} ||C^{-1/2} \Delta m||_{\mathcal{F}}^2 \big], \tag{3.4}$$

where C^{-1} is the pseudoinverse of C and $C^{-1/2}$ is the pseudoinverse of $C^{1/2}$.

As a straightforward consequence of the Feldman-Hájek theorem, we derive a closed-form expression for the KL divergence between Gaussian measures with equal covariance operators.

Proposition 1.
Let $\mathbb{P}, \mathbb{Q}, \Delta m$ be defined as in Theorem 1. Then,

$$\mathsf{KL}[\mathbb{P} \parallel \mathbb{Q}] = \frac{1}{2} \left\langle \Delta m, C^{-1} \Delta m \right\rangle_{\mathcal{F}}. \tag{3.5}$$

Proof. Appendix A.1. □

In Section 4, we make use of this result in order to develop diffusion models in function space. In Section 5, we explore various practical methods for computing this functional KL divergence under various choices of the space $\mathcal{F}$.

3.4 Gaussian Processes

Gaussian processes (GPs) (Williams and Rasmussen, 2006) are a popular class of models for specifying and learning distributions over functions. Formally, given a probability space $(\Omega, \mathcal{B}, \mathbb{P})$, a GP on $\mathcal{X}$ is a jointly measurable map $G : \Omega \times \mathcal{X} \to \mathbb{R}$ whose finite dimensional marginal distributions are Gaussian.

In practice, a Gaussian process is typically specified by a mean function $m : \mathcal{X} \to \mathbb{R}$ specifying $m(x) = \mathbb{E}[G(x)]$ and a kernel function $k : \mathcal{X}^2 \to \mathbb{R}$ specifying the covariance structure of G via $k(x, x') = \mathbb{E}[(G(x) - m(x))(G(x') - m(x'))]$. We will write $G \sim \mathcal{GP}(m, k)$ for such a Gaussian process.

Gaussian processes give us a practical way of specifying Gaussian measures, as we only need to specify a mean

function and a kernel. The kernel k plays an essential role in determining the sample path properties of a GP, such as continuity, differentiability, and periodicity (Williams and Rasmussen, 2006, Chapter 4). In the case that $m \in \mathcal{F}$ and k is chosen such that $G \in \mathcal{F}$ with probability one, we may identify G with a GRE on $\mathcal{F}$. For instance, if $m \in L^2(\mathcal{X}, \mu)$ and $\int_{\mathcal{X}} k(x,x)\,\mathrm{d}\mu(x) < \infty$, then we may identify $\mathcal{GP}(m, k)$ with a Gaussian measure on $L^2(\mathcal{X}, \mu)$. See Wild et al. (2022) and Section 5 for further details.

4 Diffusion Models in Function Space

Equipped with the necessary background, we now construct our diffusion generative model on $\mathcal{F}$. Our construction mirrors that of DDPMs (Ho et al., 2020), with the key difference being that our diffusion process takes place in a space of infinite dimensions. We note that the constructions of Ho et al. (2020) rely heavily on properties of Gaussian densities in $\mathbb{R}^n$, and thus are not directly applicable to infinite-dimensional spaces as these spaces lack a reference measure from which to define such densities (Eldredge, 2016). Note further that $\mathcal{F} = \mathbb{R}^n$ equipped with its usual inner product is a special case of our framework.

4.1 Forward Process

We begin by defining the *forward process*, a discrete-time Markov chain in $\mathcal{F}$ which iteratively perturbs our data distribution $\mathbb{P}_{\text{data}}$ towards a fixed Gaussian measure $\mathcal{N}(m, C)$. In what follows, we will choose $m = 0$ for simplicity. The choice of covariance operator C is a hyperparameter which can be tuned.

We fix a finite number of timesteps $T \in \mathbb{Z}_{>0}$ and a variance schedule $\beta : \{1, 2, \ldots, T\} \to \mathbb{R}_{>0}$, where we write β_t for $\beta(t)$. For any $u_0 \in \mathcal{F}$, we iteratively sample from the forward process via

$$u_t = \sqrt{1-\beta_t}u_{t-1} + \sqrt{\beta_t}\xi_t \qquad t = 1, 2, \ldots, T \quad (4.1)$$

where $\xi_t \sim \mathcal{N}(0, C)$ are i.i.d. Gaussian random elements on $\mathcal{F}$.

Given a fixed value of u_{t-1}, our forward process gives us conditional probability measures $\mathbb{P}_{t|t-1}(- \mid u_{t-1})$. We will write $\mathbb{P}_t$ for the marginal distribution on $\mathcal{F}$ obtained at time step t from this process, i.e.

$$\mathbb{P}_t(-) = \int_{\mathcal{F}} \mathbb{P}_{t|t-1}(- \mid u_{t-1})\,\mathrm{d}\mathbb{P}_{t-1}(u_{t-1})] \quad (4.2)$$

where $\mathbb{P}_0 = \mathbb{P}_{\text{data}}$. The value of T and the variance schedule β are chosen such that the final distribution is approximately equal to our specified Gaussian measure, i.e. $\mathbb{P}_T \approx \mathcal{N}(0, C)$.

In the following proposition, we derive expressions for several distributions related to our forward process.

Proposition 2.
Let $\gamma_t = \prod_{i=1}^{t}(1-\beta_i)$. For the forward process defined in Equation (4.1) with $m = 0$ and fixed values of u_0, u_{t-1}:

$$\mathbb{P}_{t|t-1}(- \mid u_{t-1}) = \mathcal{N}(\sqrt{1-\beta_t}u_{t-1}, \beta_t C) \quad (4.3)$$

$$\mathbb{P}_{t|0}(- \mid u_0) = \mathcal{N}(\sqrt{\gamma_t}u_0, (1-\gamma_t)C). \quad (4.4)$$

Proof. Appendix A.1. □

4.2 Reverse Process and Loss

Our generative model is then obtained by reversing the forward process, where we iteratively perturb the Gaussian measure $\mathcal{N}(0, C)$ towards the data distribution $\mathbb{P}_0$.

More specifically, to generate samples from our data distribution, we would like sample $u_T \sim \mathcal{N}(0, C)$ and iteratively sample $u_{t-1} \sim \mathbb{P}_{t-1|t}(- \mid u_t)$ from the time-reversal of our forward process for $t = T-1, \ldots, 1$. However, while the posterior probability measure $\mathbb{P}_{t-1|t}(- \mid u_t)$ is well-defined[1], it is intractable.

Most notably, using Bayes' rule here would require that the family of measures $\mathbb{P}_{t|t-1}(- \mid u_{t-1})$ be simultaneously dominated by some fixed reference measure on $\mathcal{F}$ for every choice of u_{t-1}. As these measures are Gaussian, the Feldman-Hájek theorem tells us that this is not possible (see Appendix A.1). Even if such technical difficulties were overcome (e.g. as in the Euclidean setting), computing Bayes' rule here would require computing an intractable normalization constant.

We instead take a variational approach, and approximate the posterior measures with a variational family of measures on $\mathcal{F}$ parametrized by $\theta \in \mathbb{R}^p$. In particular, we set $\mathbb{Q}_T^\theta = \mathcal{N}(0, C)$ and we approximate $\mathbb{P}_{t-1|t}(- \mid u_t)$ by the Gaussian measure

$$\mathbb{Q}_{t-1|t}^\theta(- \mid u_t) = \mathcal{N}\left(m_t^\theta(u_t), C_t^\theta(u_t)\right). \quad (4.5)$$

Here, $m_t^\theta(u_t) = m_t^\theta(- \mid u_t) \in \mathcal{F}$ is shorthand for a mean function in $\mathcal{F}$ and $C_t^\theta(u_t) = C_t^\theta(- \mid u_t) : \mathcal{F} \to \mathcal{F}$ is shorthand for a covariance operator. That is, the mean function and covariance operators depend on parameters θ as well as the timestep t and function $u_t \in \mathcal{F}$.

Although the reverse-time measures are intractable, the following proposition states that the reverse-time measures are tractable when conditioned on a starting function $u_0 \in \mathcal{F}$.

[1] This is because we assume $\mathcal{F}$ is separable, which implies that $\mathcal{F}$ is a Polish space. See Ghosal and Van der Vaart (2017, Chapter 1).

Proposition 3.
Let γ_t be defined as in Proposition (2), and consider fixed values of $u_0, u_t \in \mathcal{F}$. For $t = 2, 3, \ldots, T$, let $\widetilde{\beta}_t = \frac{1-\gamma_{t-1}}{1-\gamma_t}$ and let $\widetilde{m}_t(u_t, u_0) = \widetilde{m}_t(- \mid u_t, u_0) \in \mathcal{F}$ be defined by

$$\widetilde{m}_t(u_t, u_0) = \frac{\sqrt{\gamma_{t-1}}\beta_t}{1-\gamma_t} u_0 + \frac{\sqrt{1-\beta_t}(1-\gamma_{t-1})}{1-\gamma_t} u_t. \tag{4.6}$$

Then, $\mathbb{P}_{t-1|t,0}(- \mid u_t, u_0) = \mathcal{N}(\widetilde{m}_t(u_t, u_0), \widetilde{\beta}_t C)$.

Proof. Appendix A.1. □

We now tie our function-space Markov chain back to our observed data in order to obtain a loss function. Recall that our observations $\vec{y} \subset \mathbb{R}$ are assumed to be a vector of noisy observations of a function $u_0 \in \mathcal{F}$ at some finite collection of points $\vec{x} \subset \mathcal{X}$. We thus set the likelihood of our observed data to be $q^\theta(\mathbf{y} \mid \mathbf{x}, u_0) = \mathcal{N}(\mathbf{y}; u_0(\mathbf{x}), \sigma^2 I)$ where $\sigma^2 \in \mathbb{R}_{\geq 0}$ is some fixed constant. Note that q^θ is a Gaussian density on a finite dimensional space.

In the following proposition, we obtain a variational lower bound on the log-likelihood of our observations. This will serve as our loss function, which we seek to maximize over θ. Although this lower bound is analogous to the standard DDPM lower bound (Ho et al., 2020), the proof is non-trivial as we must work directly with the underlying probability measures rather than their densities.

Proposition 4.
The marginal likelihood of $\vec{y}$ given $\vec{x}$ is lower bounded by

$$\log q^\theta(\vec{y} \mid \vec{x}) \geq \tag{4.7}$$
$$\mathbb{E}_{\mathbb{P}}\Bigg[\log q(\vec{y} \mid \vec{x}, u_0) - \mathsf{KL}[\mathbb{P}_T(- \mid \vec{x}, \vec{y}) \parallel \mathbb{Q}_T^\theta(-)] - \sum_{t=1}^{T} \mathsf{KL}[\mathbb{P}_{t-1|t}(- \mid u_t, \vec{x}, \vec{y}) \parallel \mathbb{Q}_{t-1|t}^\theta(- \mid u_t))]\Bigg].$$

Proof. Appendix A.2. □

Since we assume $\mathbb{Q}_T^\theta$ has no trainable parameters, we may ignore the term $\mathsf{KL}[\mathbb{P}_T(- \mid \vec{x}, \vec{y}) \parallel \mathbb{Q}_T^\theta(-)]$ during training.

Mean and Covariance Parametrization We now make several further choices for our variational family. First, we analyze the terms

$$L_{t-1} = \mathsf{KL}[\mathbb{P}_{t-1|t}(- \mid u_t, u_0) \parallel \mathbb{Q}_{t-1|t}^\theta(- \mid u_t))]. \tag{4.8}$$

Note that the first measure here is Gaussian by Proposition (3), and the second is Gaussian by assumption. A more general form of the Feldman-Hájek theorem (see Appendix A.1) places strict requirements on the corresponding covariance operators in order to obtain a finite KL divergence. In particular, the term L_{t-1} will be infinite if

$$\widetilde{\beta}_t^{-1}\left(C^{-1/2}C_t^\theta(u_t)^{1/2}\right)\left(C^{-1/2}C_t^\theta(u_t)^{1/2}\right)^* - I \tag{4.9}$$

is not a Hilbert-Schmidt operator on the closure of $C^{1/2}(\mathcal{F})$. For instance, even the seemingly innocuous choice of $C_t^\theta(u_t) = \alpha\widetilde{\beta}_t C$ for any non-negative $\alpha \neq 1$ will result in an infinite KL divergence. Thus, motivated by necessity, we will choose $C_t^\theta(u_t) = \widetilde{\beta}_t C$.

Under this choice of $C_t^\theta(u_t)$, a consequence of Propositions (1) and (3) is that

$$L_{t-1} = \frac{1}{2\widetilde{\beta}_t}||C^{-1/2}(\widetilde{m}_t(u_t, u_0) - m_t^\theta(u_t))||_{\mathcal{F}}^2. \tag{4.10}$$

Similar to DDPM (Ho et al., 2020), we further choose to parametrize the mean function via

$$m_t^\theta(u_t) = \frac{1}{\sqrt{1-\beta_t}}\left(u_t - \frac{\beta_t}{\sqrt{1-\gamma_t}}\xi_t^\theta(u_t)\right) \tag{4.11}$$

where $\xi_t^\theta(u_t) \in \mathcal{F}$ is the output of a model parametrized by θ which takes in (t, u_t) as inputs and has function-valued outputs. In other words, our model is a parametrized mapping $\xi^\theta : \{1, 2, \ldots, T\} \times \mathcal{F} \to \mathcal{F}$ specified via $(t, u_t) \mapsto \xi_t^\theta(- \mid u_t)$. Under this choice, we have that

$$L_{t-1} = \lambda_t||C^{-1/2}(\xi_t - \xi_t^\theta(u_t))||_{\mathcal{F}}^2 \tag{4.12}$$

where $\lambda_t = \beta_t^2/(2\widetilde{\beta}_t(1-\beta_t)(1-\gamma_t)) \in \mathbb{R}$ is a time-dependent constant. See Appendix A.2 for details. In light of Proposition (1), we see that L_{t-1} is (up to a multiplicative constant) the KL divergence between two Gaussian measures on $\mathcal{F}$ having covariance operators C and respective means $\xi_t, \xi_t^\theta(u_t)$. As is standard in diffusion generative modeling, we drop the constant λ_t when training in order to obtain a re-weighted variational lower bound (Ho et al., 2020) for improved quality.

In Section 6, we provide a practical instantiation of the mapping ξ_t^θ via neural operators (Li et al., 2021, 2020; Kovachki et al., 2021).

Following our work, Lim et al. (2023) noted that the parametrization of the loss given in Equations 4.11 and 4.12 results in an infinite quantity when the dimension of $\mathcal{F}$ is infinite. However, it is straightforward to remedy this by considering an alternative parametrization, where the model directly predicts a rescaled version of u_0 rather than predicting ξ_t, e.g., see Appendix E and Appendix I of Lim et al. (2023) for additional details. In our experiments in this paper we used the parametrization given in Equations 4.11 and 4.12, and note that the corresponding quantities are only infinite in the limit corresponding to a discretization size of zero.

5 Function Spaces and KL Approximations

We have thus far described our framework in terms of abstract Gaussian measures on Hilbert spaces. We can obtain a concrete instantiation of our framework by choosing an appropriate space of functions to work on, as well as a choice of Gaussian measure which specifies our forward process.

In this section, we explore two choices for $\mathcal{F}$: the space of square-integrable functions $L^2(\mathcal{X}, \mu)$ and the Sobolev spaces $H^k(\mathcal{X}, \mu) = W^{k,2}(\mathcal{X}, \mu)$. We derive practical methods for estimating the KL divergence between Gaussian measures in these spaces, which is necessary for evaluating the terms in our loss function given in Equation (4.12).

To compute the functional KL divergence in Proposition (1), we derive discrete approximations of both the inverse covariance operator C^{-1} and the associated inner product. Suppose that m_1 and m_2 are known on a common discretization $\vec{x} = \{x^{(1)}, \ldots, x^{(n)}\} \subset \mathcal{X}$ which is drawn from the measure μ on $\mathcal{X}$. For any function $f : \mathcal{X} \to \mathbb{R}$, we write $f(\vec{x}) \in \mathbb{R}^n$ to represent the vector corresponding to evaluating f at the points contained in $\vec{x}$. We assume further that our Gaussian measure $\xi \sim \mathcal{GP}(0, k)$ is specified by a mean-zero Gaussian process with kernel k, with appropriate restrictions on k such that $\xi \in \mathcal{F}$ (see Section 3.4). In Appendix A.6, we explore estimating these KL divergences with spectral methods, but find that it is sensitive to the discretization size, even when the eigenfunctions are analytically known.

5.1 Square-Integrable Functions

We first consider the space $\mathcal{F} = L^2(\mathcal{X}, \mu)$ of measurable, square-integrable functions $f : \mathcal{X} \to \mathbb{R}$ equipped with the inner product $\langle f, g \rangle_{L^2} = \int_{\mathcal{X}} fg \, \mathrm{d}\mu$.

For many applications, this is a natural choice of function space as square integrability is a relatively weak assumption. Moreover, $p = 2$ is the unique choice such that the Banach space $L^p(\mathcal{X}, \mu)$ is also a Hilbert space, and the associated inner product structure is a useful tool for performing calculations.

In $L^2(\mathcal{X}, \mu)$, the covariance operator associated with our kernel function k can be explicitly described via

$$[Cg](x) = \int_{\mathcal{X}} k(x, x') g(x') \, \mathrm{d}\mu(x') \qquad \forall g \in \mathcal{F}. \quad (5.1)$$

We provide a derivation of this formula for C in Appendix A.3. Let $K_{\vec{x}\vec{x}} \in \mathbb{R}^{n \times n}$ be the covariance matrix specified by k and evaluated on $\vec{x}$, i.e. the (i, j)th entry of $K_{\vec{x}\vec{x}}$ is given by $k(x_i, x_j)$. Then, upon replacing μ with the empirical measure specified by $\vec{x}$, we have $[Cg](\vec{x}) \approx n^{-1} K_{\vec{x}\vec{x}} g(\vec{x}) \in \mathbb{R}^n$, so that the (scaled) covariance matrix K is a discrete approximation of the covariance operator C which may inverted. Replacing μ once more with the empirical measure specified by $\vec{x}$, we then have

$$\mathsf{KL}[\mathcal{N}(m_1, C) \parallel \mathcal{N}(m_2, C)] \approx \frac{1}{2} \Delta m(\vec{x})^T K_{\vec{x}\vec{x}}^{-1} \Delta m(\vec{x}). \quad (5.2)$$

Interestingly, this is precisely the KL divergence between two finite-dimensional Gaussians with equal covariance matrices $K_{\vec{x}\vec{x}}$ and means $m_1(\vec{x}), m_2(\vec{x})$.

Furthermore, we note that Sun et al. (2019) prove that the KL divergence between two stochastic processes is the supremum of the KL divergences between their finite-dimensional marginals. Our approximation in Equation (5.2) is increasing under refinements of the observation set $\vec{x}$, and thus is a lower bound on the true KL divergence.

Proposition 5.
Equation (5.2) is strictly increasing under refinements of the observation set $\vec{x}$. In particular, if $\vec{z} \subset \vec{x}$, then

$$\Delta m(\vec{z})^T K_{\vec{z}\vec{z}}^{-1} \Delta m(\vec{z}) \leq \Delta m(\vec{x})^T K_{\vec{x}\vec{x}}^{-1} \Delta m(\vec{x}). \quad (5.3)$$

Proof. Appendix A.3. □

5.2 Sobolev Spaces

A second choice of function spaces that have many practical applications are the Sobolev spaces $H^k(\mathcal{X}, \mu)$ consisting of functions in $L^2(\mathcal{X}, \mu)$ whose mixed αth-order partial derivatives of order at most k exist (in a weak sense) and are also in $L^2(\mathcal{X}, \mu)$ (Evans, 2010, Chapter 5). Of particular interest is the setting where $\mathcal{X} \subset \mathbb{R}$ and $k = 1$, where the inner product is given by

$$\langle f, g \rangle_{H^1} = \langle f, g \rangle_{L^2} + \langle \partial_x f, \partial_x g \rangle_{L^2}. \quad (5.4)$$

When the Gaussian process associated with the kernel function k lies in H^1 with probability one, the corresponding covariance operator can be expressed as

$$[Cg](x) = \int_{\mathcal{X}} k(x, x') \, \mathrm{d}\mu(x') + \int_{\mathcal{X}} \left[\partial_{x'} k(x, x')\right] \left[\partial_{x'} g(x')\right] \, \mathrm{d}\mu(x'). \quad (5.5)$$

See Appendix A.3 for a derivation. Our discretization in this setting follows closely that of our techniques for the space $L^2(\mathcal{X}, \mu)$, with the additional necessity of employing a discrete differential operator. To that end, let $D \in \mathbb{R}^{n \times n}$ be any discrete approximation to the first-order differentiation operator. In practice we use a discretization based on finite-difference equations. Let $K'_{\vec{x}\vec{x}} \in \mathbb{R}^{n \times n}$ be the covariance matrix corresponding to the differeniated kernel $\partial_{x'} k(x, x')$. That is, the (i, j)th entry of $K'_{\vec{x}\vec{x}}$ is given by $\frac{\partial}{\partial x'} k(x_i, x_j)$. Then, the covariance operator C can be

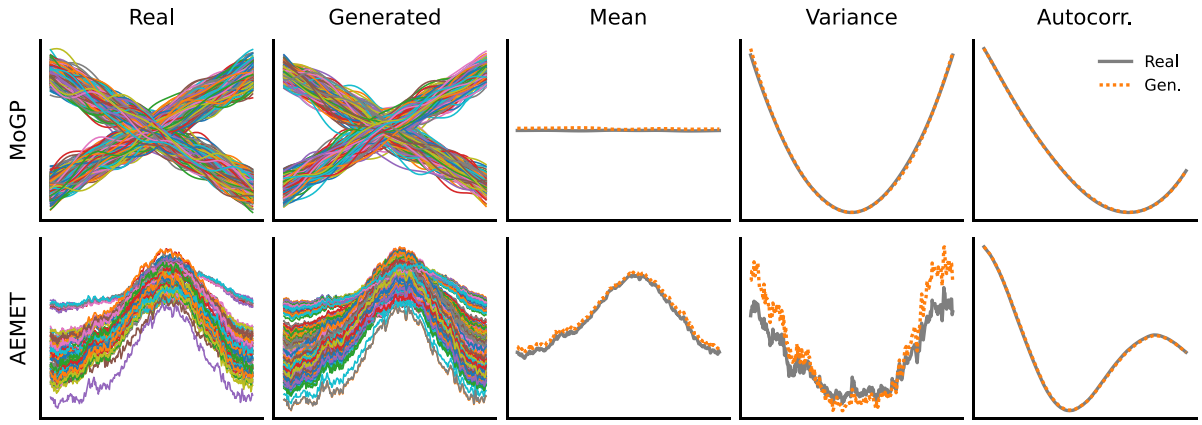

Figure 1: Unconditional function generation on a synthetic (MoGP) and real-world (AEMET) dataset. For each dataset, a GNO model was trained on the plotted functions (first column), and a total of 500 functions were sampled from the model (second column). The generated curves closely match the training curves in both perceptual quality and pointwise statistics.

discretized via $[Cg](\vec{x}) \approx n^{-1}\left[K_{\vec{x}\vec{x}} + K'_{\vec{x}\vec{x}}D\right] g(\vec{x}) \in \mathbb{R}^n$, and moreover,

$$\mathsf{KL}[\mathcal{N}(m_1, C) \,\|\, \mathcal{N}(m_2, C)] \approx \frac{1}{2}\Delta m(\vec{x})^T \left[I + D^T D\right] \left[K_{\vec{x}\vec{x}} + K'_{\vec{x}\vec{x}}D\right]^{-1} \Delta m(\vec{x}). \tag{5.6}$$

Although the covariance operator C is guaranteed to be positive semidefinite in theory, discretizing this operator often results in a non-PSD matrix approximation which may cause training to diverge. In practice, we project the matrix $[I + D^T D][K_{\vec{x}\vec{x}} + K'_{\vec{x}\vec{x}}D]^{-1}$ to the nearest symmetric PSD matrix (in terms of the Frobenius norm) (Higham, 1988; Cheng and Higham, 1998). See Appendix A.3 for details.

5.3 Existing Methods in Terms of Our Theory

In terms of our methodology, existing methods (Kong et al., 2021; Chen et al., 2021; Tashiro et al., 2021; Dutordoir et al., 2022) can be viewed as operating in the space $\mathcal{F} = L^2(\mathcal{X}, \mu)$, with the discretization employed in Equation (5.2). In all of these methods, the forward process is defined via a white noise prior. However, such a prior can *not* be seen as a Gaussian measure. In particular, the white noise process is not jointly measurable (Kallianpur, 2013, Example 1.2.5), and thus one is unable to consider the corresponding sample paths as elements of some function space. A GRE corresponding to this prior would have infinite variance, as the corresponding covariance operator would be the identity operator. Nonetheless, despite these foundational concerns, existing methods show strong empirical performance. Explaining this performance from a functional point of view, for example through the theory of generalized functions (Grubb, 2008), is an interesting challenge for future work.

6 Experiments

In this section, we perform several experiments in order to illustrate how our theoretical framework can be implemented as a practical estimation methodology. In all experiments, we parametrize $\xi_t^\theta(u_t)$ via a graph neural operator (GNO) (Li et al., 2020; Kovachki et al., 2021). See Appendix A.4 for our model configurations and hyperparameter settings. Our models are trained by minimizing the reweighted negative ELBO as described in Section 4. In all plots, our functional diffusion model is denoted *FuncDiff*. Pseudocode and additional details for all of our algorithms is available in Appendix A.5.

Code for all of our experiments is available at `https://github.com/GavinKerrigan/functional_diffusion`.

A key property of the GNO is the ability to condition on arbitrary discretizations of $\mathcal{X}$. This allows us to train our models on functions that are observed at different points, as well as to condition on arbitrary function observations when performing conditional generation. Moreover, as neural operators parametrize mappings between function spaces, we are able to query our model at arbitrary input locations. Thus, our model is not tied to any particular discretization.

Datasets We use both a synthetic and a real-world dataset in the main paper to illustrate our approach, with results on additional real-world datasets in Appendix A.6. Our synthetic dataset is a mixture of Gaussian processes (*MoGP*) with a squared-exponential kernel with variance $\sigma^2 = 0.4$ and length scale $\ell = 0.1$, where the first mixture component has mean $m_1 = 10x - 5$ and the second has mean $m_2 = -10x + 5$. These functions are observed on a uniform discretization of $[0, 1] \subset \mathbb{R}$. We use 64 observation

points unless otherwise specified. Our real-world dataset (*AEMET*) is a well-known dataset in the functional data analysis literature. This dataset consists of 73 curves, where each curve is the mean daily temperature at a particular Spanish weather station, so that each curve has a total of 365 discrete observations (Febrero-Bande and de la Fuente, 2012). See Figure 1 for an illustration of these datasets.

6.1 Unconditional Generation

In this experiment, we sample curves unconditionally from our trained model. In Figure 1, the generated curves closely match the training data in terms of perceptual qualities. We additionally compute the pointwise mean, pointwise variance, and mean autocorrelation of both the real and generated curves. The summary statistics of the generated data closely match those of the real data, indicating that the model has successfully learned to sample from the functional distribution. See Appendix A.6 for a comparison to a simple baseline based on functional PCA (Ramsay and Silverman, 2008, Chapter 6) and additional datasets.

6.2 Conditional Generation

Our proposed approach for conditional generation is an extension of the ILVR method (Choi et al., 2021) to functional data. This method works by perturbing conditioning information via the forward process, and during generation we set the values of the generated function at the conditioning locations to these perturbed values. In particular note that we are able to condition a pre-trained unconditional model on arbitrary function observations. Thus, this method may potentially be applied to a wide array of tasks, such as extrapolation, upsampling, or data imputation.

In Figure 2, we demonstrate this by conditioning our generation on a known segment of the function. We see that our method is able to leverage the learned functional distribution in order to accurately extrapolate the given conditioning information. We compare to a Gaussian process regression (*GPR*) baseline, where we fit a Gaussian process only to the conditioning information. Unsurprisngly, the GPR method is not able to accurately extrapolate the conditioning information, as it has no additional information regarding the underlying functional distribution.

Moreover, our conditioning method allows us to do *soft conditioning*, where the diffusion process is not conditioned on the observed values for some number of the final diffusions steps. This allows us to generate curves that are similar to a given observation, but not exactly matching. For example, this can be used to select a particular mode to sample from in a multimodal dataset. We demonstrate this in Figure 3. As a potential future application, soft conditioning could be applied as a data augmentation method for functional data.

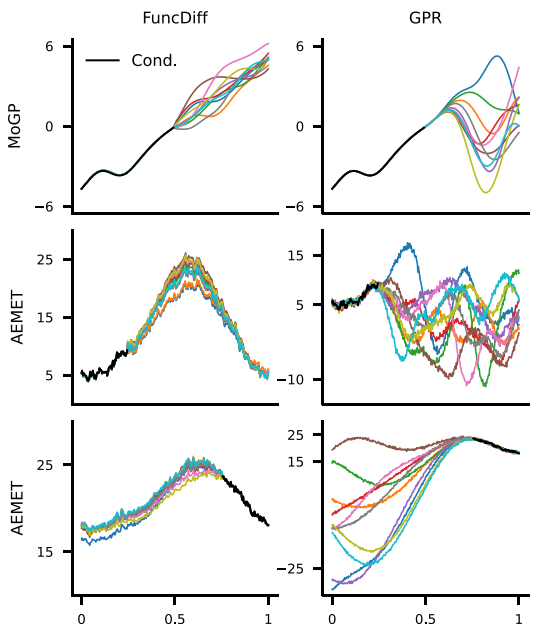

Figure 2: Conditional samples of our model (FuncDiff) are compared against Gaussian process regression (GPR). In each plot, both models are conditioned on the black curves.

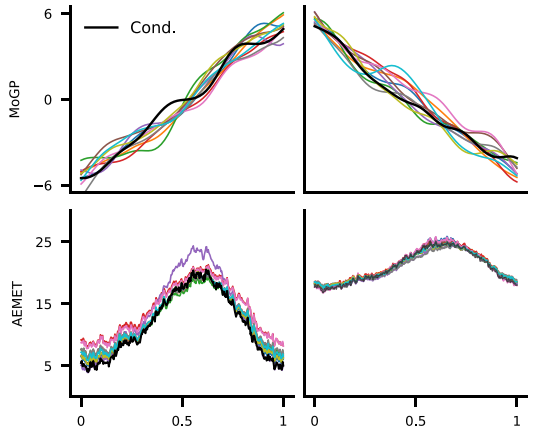

Figure 3: An illustration of our soft conditioning method. We condition the generative process on the black curves for all but the final 150 diffusion steps. This allows us to generate functions that are qualitatively similar to the given conditioning information (in black), such that the generated function values do not necessarily exactly match those of the conditioning information.

6.3 Function Spaces

Lastly, we experiment with the choice of function space. In particular, we compare the use of the $L^2(\mathcal{X}, \mu)$ inner product against the use of the $H^1(\mathcal{X}, \mu)$ inner product. Intuitively, the derivative term in Sobolev inner product will

penalize generated functions that are not smooth. In Table 2, we measure the smoothness of generated curves by computing the mean standard deviation of the derivatives of said curves. We find empirically that using the Sobolev loss can result in significantly smoother generations when the underlying functional dataset is highly regular. As smoothness is not a desirable property for the AEMET dataset, we include here a dataset consisting of linear functions (*Linear*) instead. See Appendix A.6 for more on this dataset. We use the Matérn kernel with $\nu = 3/2$ when working with the the Sobolev norm as this kernel has differentiable sample paths.

Table 1: Mean smoothness of generated functions as measured by the standard deviation of the function derivatives, averaged across 500 samples. Using the Sobolev norm over the L^2 norm can significantly increase the smoothness of generated functions, while not harming performance if the generated functions are already sufficiently smooth.

Dataset	$L^2(\mathcal{X}, \mu)$	$H^1(\mathcal{X}, \mu)$
Linear	0.753	0.203
MoGP	24.73	24.74

7 Conclusion

We propose a framework for diffusion generative modeling in infinite-dimensional spaces of functions and develop practical techniques for realizing this framework on real-world data. Enabled by our framework, future functional diffusion models may be able move beyond the typical L^2-space assumption in order to incorporate informative prior information. A remaining challenge for functional diffusion models is to consider the continuous-time limit and elucidating connections with score-based methods (Song et al., 2021). For instance, it may be possible to view continuous-time functional diffusion models as stochastic PDEs, potentially enabling more efficient sampling methods.

Acknowledgements

This research was supported in part by the National Science Foundation under award number 1900644, by the HPI Research Center in Machine Learning and Data Science at UC Irvine, and by a Qualcomm Faculty award.

References

Juan Lopez Alcaraz and Nils Strodthoff. Diffusion-based time series imputation and forecasting with structured state space models. *Transactions on Machine Learning Research*, 2022.

Marin Biloš, Kashif Rasul, Anderson Schneider, Yuriy Nevmyvaka, and Stephan Günnemann. Modeling temporal data as continuous functions with process diffusions. In *Workshop on Score-Based Methods*, 2022.

Vladimir Igorevich Bogachev. *Gaussian Measures*. American Mathematical Society, 1998.

Valentin De Bortoli, Emile Mathieu, Michael John Hutchinson, James Thornton, Yee Whye Teh, and Arnaud Doucet. Riemannian score-based generative modelling. In *Advances in Neural Information Processing Systems*, 2022.

David R Burt. Spectral methods in Gaussian process approximations. Master's thesis, University of Cambridge, 2018.

David R. Burt, Sebastian W. Ober, Adrià Garriga-Alonso, and Mark van der Wilk. Understanding variational inference in function-space. In *Third Symposium on Advances in Approximate Bayesian Inference*, 2021.

Nanxin Chen, Yu Zhang, Heiga Zen, Ron J Weiss, Mohammad Norouzi, and William Chan. Wavegrad: Estimating gradients for waveform generation. In *International Conference on Learning Representations*, 2021.

Sheung Hun Cheng and Nicholas J Higham. A modified Cholesky algorithm based on a symmetric indefinite factorization. *SIAM Journal on Matrix Analysis and Applications*, 19(4):1097–1110, 1998.

Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. ILVR: Conditioning method for denoising diffusion probabilistic models. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 14347–14356, 2021.

Giuseppe Da Prato and Jerzy Zabczyk. *Stochastic Equations in Infinite Dimensions*. Cambridge University Press, 2014.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems*, pages 8780–8794, 2021.

Emilien Dupont, Hyunjik Kim, S. M. Ali Eslami, Danilo Jimenez Rezende, and Dan Rosenbaum. From data to functa: Your data point is a function and you can treat it like one. In *Proceedings of the 39th International Conference on Machine Learning*, pages 5694–5725, 2022a.

Emilien Dupont, Yee Whye Teh, and Arnaud Doucet. Generative models as distributions of functions. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, pages 2989–3015, 2022b.

Paul Dupuis and Richard S Ellis. *A weak convergence approach to the theory of large deviations*. John Wiley & Sons, 2011.

Vincent Dutordoir, Alan Saul, Zoubin Ghahramani, and Fergus Simpson. Neural diffusion processes. *arXiv preprint arXiv:2206.03992*, 2022.

Nathaniel Eldredge. Analysis and probability on infinite-dimensional spaces. *arXiv preprint arXiv:1607.03591*, 2016.

Lawrence C Evans. *Partial Differential Equations*. Graduate Studies in Mathematics. American Mathematical Society, 2010.

Manuel Febrero-Bande and Manuel Oviedo de la Fuente. Statistical computing in functional data analysis: The R package fda.usc. *Journal of statistical Software*, 51:1–28, 2012.

Marta Garnelo, Jonathan Schwarz, Dan Rosenbaum, Fabio Viola, Danilo J Rezende, SM Eslami, and Yee Whye Teh. Neural processes. In *Workshop on Theoretical Foundations and Applications of Deep Generative Models*, 2018.

Subhashis Ghosal and Aad Van der Vaart. *Fundamentals of Nonparametric Bayesian Inference*. Cambridge University Press, 2017.

Gerd Grubb. *Distributions and Operators*. Springer Science & Business Media, 2008.

Nicholas J. Higham. Computing a nearest symmetric positive semidefinite matrix. *Linear Algebra and its Applications*, 103:103–118, 1988.

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

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022.

Chin-Wei Huang, Milad Aghajohari, Joey Bose, Prakash Panangaden, and Aaron Courville. Riemannian diffusion models. In *Advances in Neural Information Processing Systems*, 2022.

Gopinath Kallianpur. *Stochastic Filtering Theory*. Springer Science & Business Media, 2013.

Hyunjik Kim, Andriy Mnih, Jonathan Schwarz, Marta Garnelo, Ali Eslami, Dan Rosenbaum, Oriol Vinyals, and Yee Whye Teh. Attentive neural processes. In *International Conference on Learning Representations*, 2019.

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

Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces. *arXiv preprint arXiv:2108.08481*, 2021.

Olivier Le Maître and Omar M Knio. *Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics*. Springer Science & Business Media, 2010.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Graph kernel network for partial differential equations. *arXiv preprint arXiv:2003.03485*, 2020.

Zongyi Li, Nikola Borislavov Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021.

Jae Hyun Lim, Nikola B. Kovachki, Ricardo Baptista, Christoper Beckham, Kamyar Azizzadenesheli, Jean Kossaifi, Vikram Voleti, Jiaming Song, Karsten Kreis, Jan Kautz, Christopher Pal, Arash Vahdat, and Anima Anandkumar. Score-based diffusion models in function space. *arXiv preprint arXiv:2302.07400*, 2023.

Alexander G. de G. Matthews, James Hensman, Richard Turner, and Zoubin Ghahramani. On sparse variational methods and the Kullback-Leibler divergence between stochastic processes. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 231–239, 2016.

Athansios Papoulis and S. Unnikrishna Pillai. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, 2002.

Jakiw Pidstrigach, Youssef Marzouk, Sebastian Reich, and Sven Wang. Infinite-dimensional diffusion models for function spaces. *arXiv preprint arXiv:2302.10130*, 2023.

Md Ashiqur Rahman, Manuel A Florez, Anima Anandkumar, Zachary E Ross, and Kamyar Azizzadenesheli. Generative adversarial neural operators. *Transactions on Machine Learning Research*, 2022.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with CLIP latents. *arXiv preprint arXiv:2204.06125*, 2022.

Carlos Ramos-Carreño, Alberto Suárez, José Luis Torrecilla, Miguel Carbajo Berrocal, Pablo Marcos Manchón, Pablo Pérez Manso, Amanda Hernando Bernabé, David García Fernández, Yujian Hong, Pedro Martín Rodríguez-Ponga Eyriès, Álvaro Sánchez Romero, Elena Petrunina, Álvaro Castillo, Diego Serna, and Rafael Hidalgo. GAA-UAM/scikit-fda: Functional data analysis in Python, 2019.

James O. Ramsay and Bernhard W. Silverman. *Functional Data Analysis*. Springer New York, 2008.

Kashif Rasul, Calvin Seward, Ingmar Schuster, and Roland Vollgraf. Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8857–8868, 2021.

Tim G. J. Rudner, Zonghao Chen, and Yarin Gal. Rethinking function-space variational inference in Bayesian neural networks. In *Third Symposium on Advances in Approximate Bayesian Inference*, 2021.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in Neural Information Processing Systems*, 2022.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 2256–2265, 2015.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, pages 11918—-11930, 2019.

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

Shengyang Sun, Guodong Zhang, Jiaxin Shi, and Roger Grosse. Functional variational Bayesian neural networks. In *International Conference on Learning Representations*, 2019.

Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. CSDI: Conditional score-based diffusion models for probabilistic time series imputation. In *Advances in Neural Information Processing Systems*, pages 24804–24816, 2021.

Ba-Hien Tran, Simone Rossi, Dimitrios Milios, and Maurizio Filippone. All you need is a good functional prior for bayesian deep learning. *Journal of Machine Learning Research*, 23(74):1–56, 2022.

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

Veit D. Wild, Robert Hu, and Dino Sejdinovic. Generalized variational inference in function spaces: Gaussian measures meet Bayesian deep learning. In *Advances in Neural Information Processing Systems*, 2022.

Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

George Wynne and Veit Wild. Variational gaussian processes: a functional analysis view. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, pages 4955–4971, 2022.

Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. Geodiff: A geometric diffusion model for molecular conformation generation. In *International Conference on Learning Representations*, 2022.

Ruihan Yang, Prakhar Srivastava, and Stephan Mandt. Diffusion probabilistic modeling for video generation. *arXiv preprint arXiv:2203.09481*, 2022.

Diffusion Generative Models in Infinite Dimensions: Supplementary Materials

Summary of the Appendix

The appendix is organized as follows:

- In Section A.1, we discuss additional properties of Gaussian measures not contained in the main paper. In particular, we show that the space of Gaussian measures is closed under affine transformations and independent sums. We leverage these facts to prove Propositions (2), (3). In addition, we provide a statement of the general formulation of the Feldman-Hájek theorem, and use this theorem to prove Proposition (1).
- In Section A.2, we provide a proof of the ELBO in Proposition (4), as well as a detailed derivation of the reparametrization and simplified loss of Section 4.
- In Section A.3, we derive expressions for the covariance operator associated with a Gaussian process under the assumptions that $\mathcal{F} = L^2(\mathcal{X}, \mu)$ or $\mathcal{F} = H^1(\mathcal{X}, \mu)$. In addition, we prove Proposition (5).
- In Section A.4, we provide the details of the model architectures and training procedures used in our experiments, as well as an ablation on the choice of kernel in the forward process.
- In Section A.5 we provide pseudocode for training our model as well as both unconditional and conditional sampling.
- In Section A.6, we include additional experiments not detailed in the main paper. These include unconditional generation results on additional datasets, a comparison to a simple baseline based on functional PCA, and a comparison between the discrete approximations of the KL divergence proposed in Section 5 to a method based on computing the spectrum of the corresponding covariance operator.

A.1 Gaussian Measures

In this section, we include some useful facts and additional details regarding Gaussian measures, as well as proofs of Propositions (1), (2), (3).

A.1.1 Basic Properties

Lemma 1 (Affine Transformations of GREs)**.**
Let $F \sim \mathcal{N}(m, C)$ be a GRE on $\mathcal{F}$. Then, for $\alpha \in \mathbb{R}$ and $g \in \mathcal{F}$, we have that $\alpha F + g \sim \mathcal{N}(\alpha m + g, \alpha^2 C)$.

Proof. Fix any $h \in \mathcal{F}$, and note that

$$\langle h, \alpha F + g\rangle = \alpha\langle h, F\rangle + \langle h, g\rangle. \tag{A.1.1}$$

Since $\langle h, F\rangle \sim \mathcal{N}(\langle h, m\rangle, \langle Ch, h\rangle)$, it follows that $\langle h, \alpha F + g\rangle$ must follow a Gaussian distribution on $\mathbb{R}$ with mean

$$\alpha\langle h, m\rangle + \langle h, g\rangle = \langle h, \alpha m + g\rangle \tag{A.1.2}$$

and variance

$$\alpha^2\langle Ch, h\rangle = \langle \alpha^2 Ch, h\rangle. \tag{A.1.3}$$

Thus, we have shown that $\alpha F + g$ is a GRE on $\mathcal{F}$, as its inner product with arbitrary $h \in \mathcal{F}$ is Gaussian on $\mathbb{R}$. Moreover, we have computed its mean and covariance operator as claimed. □

Lemma 2 (Sum of Independent GREs)**.**
If $F \sim \mathcal{N}(m_1, C_1)$ and $G \sim \mathcal{N}(m_2, C_2)$ are independent GREs on $\mathcal{F}$, then $F + G \sim \mathcal{N}(m_1 + m_2, C_1 + C_2)$.

Proof. Let $Z = F + G$. Write $\mathbb{P}_F, \mathbb{P}_G, \mathbb{P}_Z$ for the probability measures of F, G, Z respectively. The Fourier transform of $\mathbb{P}_F$ is given by

$$\widehat{\mathbb{P}}_F(\lambda) = \int_{\mathcal{F}} \exp\left[i\langle \lambda, F\rangle\right] \mathrm{d}\mathbb{P}_F \qquad \forall \lambda \in \mathcal{F}, \tag{A.1.4}$$

and is given analogously for our other measures. By Bogachev (1998, A.3.17) and the subsequent discussion, a probability measure is uniquely determined by its Fourier transform. Moreover, we have that

$$\widehat{\mathbb{P}}_Z(\lambda) = \widehat{\mathbb{P}}_F(\lambda)\widehat{\mathbb{P}}_G(\lambda) \qquad \forall \lambda \in \mathcal{F}. \tag{A.1.5}$$

Using the expression for the Fourier transform of a Gaussian measure given in Da Prato and Zabczyk (2014, Chapter 2), we see that that

$$\hat{P}_Z(\lambda) = \exp\left[i\langle \lambda, m_1\rangle - \frac{1}{2}\langle C_1\lambda, \lambda\rangle\right] \exp\left[i\langle \lambda, m_2\rangle - \frac{1}{2}\langle C_2\lambda, \lambda\rangle\right] \tag{A.1.6}$$

$$= \exp\left[i\langle \lambda, m_1 + m_2\rangle - \frac{1}{2}\langle (C_1 + C_2)\lambda, \lambda\rangle\right] \tag{A.1.7}$$

which is precisely the Fourier transform of the measure $\mathcal{N}(m_1 + m_2, C_1 + C_2)$.

□

A.1.2 Diffusion Process Measures

In this subsection we derive various closed-form measures related to our diffusion process in Section (4).

Proof of Proposition (2)**.**

Proof. The first and second claims are special cases of Lemma (1).

For the third claim, we proceed by induction on t. The case $t = 1$ is clear from Lemma (1). Now, suppose

$$u_{t-1} \mid u_0 \sim \mathcal{N}(\sqrt{\gamma_{t-1}}u_0, (1 - \gamma_{t-1})C). \tag{A.1.8}$$

By the definition of the forward process and our inductive assumption, we have that $u_t = \sqrt{1-\beta_t}u_{t-1} + \sqrt{\beta_t}\xi_t$ is the sum of two independent GREs: the first is

$$\sqrt{1-\beta_t}u_{t-1} \sim \mathcal{N}(\sqrt{\gamma_t}u_0, (1-\beta_t)(1-\gamma_{t-1})C) \tag{A.1.9}$$

and the second is $\sqrt{\beta_t}\xi_t \sim \mathcal{N}(0, \beta_t C)$. By Lemma (2), we obtain the result, as

$$(1-\beta_t)(1-\gamma_{t-1}) + \beta_t = 1 - (1-\beta_t)\gamma_{t-1} = 1 - \gamma_t. \tag{A.1.10}$$

□

Proof of Proposition (3)**.**

Proof. By Proposition (2) and Lemma (1), we may write

$$u_{t-1} = \sqrt{\gamma_{t-1}}u_0 + \sqrt{1-\gamma_{t-1}}\xi \qquad \xi \sim \mathcal{N}(0, C) \tag{A.1.11}$$

and by construction we have

$$u_t = \sqrt{1-\beta_t}u_{t-1} + \sqrt{\beta_t}\xi' \qquad \xi' \sim \mathcal{N}(0, C) \tag{A.1.12}$$

where $\xi, \xi' \sim \mathcal{N}(0, C)$ are independent GREs. Our strategy is to manipulate these expressions to obtain a reparametrized expression for u_{t-1}. By Equation (A.1.11),

$$\beta_t\sqrt{\gamma_{t-1}}u_0 = \beta_t\left[u_{t-1} - \sqrt{1-\gamma_{t-1}}\xi\right], \tag{A.1.13}$$

and similarly by Equation (A.1.12),

$$(1-\gamma_{t-1})\sqrt{1-\beta_t}u_t = (1-\gamma_{t-1})\left[(1-\beta_t)u_{t-1} + \sqrt{\beta_t}\sqrt{1-\beta_t}\xi'\right]. \tag{A.1.14}$$

Upon summing Equations (A.1.13)-(A.1.14) and isolating the u_{t-1} terms,

$$\begin{aligned}(\beta_t + (1-\gamma_{t-1})(1-\beta_t))\,u_{t-1} = \beta_t\sqrt{\gamma_{t-1}}u_0 + (1-\gamma_{t-1})\sqrt{1-\beta_t}u_t \\ + \beta_t\sqrt{1-\gamma_{t-1}}\xi - (1-\gamma_{t-1})\sqrt{\beta_t}\sqrt{1-\beta_t}\xi'.\end{aligned} \tag{A.1.15}$$

On the LHS, we have

$$\begin{aligned}(\beta_t + (1-\gamma_{t-1})(1-\beta_t))u_{t-1} &= (\beta_t + (1-\beta_t) - (1-\beta_t)(\gamma_{t-1}))u_{t-1} \\ &= (1-\gamma_t)u_{t-1}\end{aligned} \tag{A.1.16}$$

thereby allowing us to obtain

$$u_{t-1} = \frac{\beta_t\sqrt{\gamma_{t-1}}}{1-\gamma_t}u_0 + \frac{\sqrt{1-\beta_t}(1-\gamma_{t-1})}{1-\gamma_t}u_t \tag{A.1.17}$$

$$+ \frac{\beta_t\sqrt{1-\gamma_{t-1}}}{1-\gamma_t}\xi - \frac{(1-\gamma_{t-1})\sqrt{\beta_t}\sqrt{1-\beta_t}}{1-\gamma_t}\xi'. \tag{A.1.18}$$

We now analyize the noise terms (i.e. only those terms depending on ξ, ξ'). By Lemmas (1)-(2) and the independence of ξ, ξ', the sum of the noise terms follows a mean zero Gaussian measure with covariance

$$\begin{aligned}&\left(\frac{\beta_t\sqrt{1-\gamma_{t-1}}}{1-\gamma_t}\right)^2 + \left(\frac{(1-\gamma_{t-1})\sqrt{\beta_t}\sqrt{1-\beta_t)}}{1-\gamma_t}\right)^2 \\ &= \left(\frac{\beta_t(1-\gamma_{t-1})}{1-\gamma_t}\right)\left(\frac{\beta_t + (1-\beta_t)(1-\gamma_{t-1})}{1-\gamma_t}\right) \\ &= \frac{\beta_t(1-\gamma_{t-1})}{1-\gamma_t}.\end{aligned}$$

where the last line follows from the calculation in Equation (A.1.16). Thus, we see that $u_{t-1} \mid u_t, u_0$ follows a Gaussian measure with the claimed mean and covariance. □

A.1.3 The Feldman-Hájek Theorem and its Consequences

Here we state the Feldman-Hájek Theorem in its general form, and discuss a few of its consequences. We include here only a statement of the theorem – see Da Prato and Zabczyk (2014, Theorem 2.23, Theorem 2.25) for a proof.

Theorem 2 (The Feldman-Hájek Theorem, General Case)**.**
Let $\mathbb{P} = \mathcal{N}(m_1, C_1)$ and $\mathbb{Q} = \mathcal{N}(m_2, C_2)$ be Gaussian measures on $\mathcal{F}$. Then,

1. The measures $\mathbb{P}$ and $\mathbb{Q}$ are either equivalent (i.e. $\mathbb{P} \ll \mathbb{Q}$ and $\mathbb{Q} \ll \mathbb{P}$) or mutually singular (i.e. $\mathbb{P}$ is not absolutely continuous with respect to $\mathbb{Q}$ and vice-versa).
2. The measures $\mathbb{P}$ and $\mathbb{Q}$ are equivalent if and only if:
   (a) $C_1^{1/2}(\mathcal{F}) = C_2^{1/2}(\mathcal{F}) = H_0$
   (b) $m_1 - m_2 \in H_0$
   (c) The operator $(C_1^{-1/2}C_2^{1/2})(C_1^{-1/2}C_2^{1/2})^* - I$ is Hilbert-Schmidt on the closure $\overline{H_0}$.
3. If $\mathbb{P}$ and $\mathbb{Q}$ are equivalent and $C_1 = C_2 = C$, then $\mathbb{Q}$-a.s. the Radon-Nikodym derivative $\mathrm{d}\mathbb{P}/\,\mathrm{d}\mathbb{Q}$ is given by

$$\frac{\mathrm{d}\mathbb{P}}{\mathrm{d}\mathbb{Q}}(f) = \exp\left[\langle C^{-1/2}(m_1 - m_2), C^{-1/2}(f - m_2)\rangle - \frac{1}{2}||C^{-1/2}m_1 - m_2)||^2\right] \qquad \forall f \in \mathcal{F} \tag{A.1.19}$$

An important consequence of the Feldman-Hájek theorem that we repeatedly make use of throughout our work is that it allows is to compute the KL between Gaussian measures having equal covariance operators.

Proof of Proposition (1)**.**

Proof. Suppose $\mathbb{P} = \mathcal{N}(m_1, C)$ and $\mathbb{Q} = \mathcal{N}(m_2, C)$ and that $m_1 - m_2 \in C^{1/2}(\mathcal{F})$. It follows from the Feldman-Hájek theorem that $\mathbb{P}$ and $\mathbb{Q}$ are equivalent. We now use the Radon-Nikodym expression from the Feldman-Hájek theorem to compute the KL divergence.

We have that

$$\begin{aligned}
\mathsf{KL}[\mathbb{P} \parallel \mathbb{Q}] &= \int_{\mathcal{F}} \log \frac{\mathrm{d}\mathbb{P}}{\mathrm{d}\mathbb{Q}}(f)\,\mathrm{d}\mathbb{P}(f) && \text{(A.1.20)}\\
&= -\frac{1}{2}||C^{-1/2}(m_1 - m_2))|| + \int_{\mathcal{F}} \langle C^{-1/2}(m_1 - m_2), C^{-1/2}(f - m_2)\,\mathrm{d}\mathbb{P}(f). && \text{(A.1.21)}
\end{aligned}$$

We now analyze the integral term via a spectral decomposition. Let $\{(\lambda_j, e_j)\}_{j=1}^{\infty}$ be the eigenvalues and eigenvectors of C. Note that the eigenvectors of C form an orthonormal basis for $\mathcal{F}$ by the spectral theorem, as C is a self-adjoint compact operator. Then, we may evaluate the second integral as

$$\begin{aligned}
&\int_{\mathcal{F}} \langle C^{-1/2}(m_1 - m_2), C^{-1/2}(f - m_2)\,\mathrm{d}\mathbb{P}(f) && \text{(A.1.22)}\\
&= \int_{\mathcal{F}} \sum_{j=1}^{\infty} \langle m_1 - m_2, e_j\rangle\langle f - m_2, e_j\rangle \lambda_j^{-1}\,\mathrm{d}\mathbb{P}(f) && \text{(A.1.23)}\\
&= \sum_{j=1}^{\infty} \lambda_j^{-1}\langle m_1 - m_2, e_j\rangle \int_{\mathcal{F}} \langle f - m_2, e_j\rangle\,\mathrm{d}\mathbb{P}(f) && \text{(A.1.24)}\\
&= \sum_{j=1}^{\infty} \lambda_j^{-1}\langle m_1 - m_2, e_j\rangle^2 && \text{(A.1.25)}\\
&= \langle C^{-1/2}(m_1 - m_2), C^{-1/2}(m_1 - m_2)\rangle. && \text{(A.1.26)}
\end{aligned}$$

Combining this computation with the KL expression above completes the proof. □

A.2 Loss Function

In this section we provide additional details regarding the derivation and parametrization of our loss function.

A.2.1 Functional ELBO

Proof of Proposition (4)**.**

Proof. First, we apply the usual functional ELBO (Wild et al., 2022; Matthews et al., 2016; Sun et al., 2019), treating $u_{0:T}$ as latent variables and using the assumption that the reverse-time chain is Markov to obtain

$$\log q^\theta(\vec{y} \mid \vec{x}) \geq \mathbb{E}_{\mathbb{P}}\left[\log q^\theta(\vec{y} \mid \vec{x}, u_0)\right] - \mathsf{KL}[\mathbb{P}(\,\mathrm{d}u_{0:T} \mid \vec{x}, \vec{y}) \parallel \mathbb{Q}^\theta(\,\mathrm{d}u_{0:T})]. \tag{A.2.1}$$

By the chain rule for KL divergences (Dupuis and Ellis, 2011), we may condition on u_T to obtain

$$\begin{aligned}\log q^\theta(\vec{y} \mid \vec{x}) \geq \mathbb{E}_{\mathbb{P}}\left[\log q^\theta(\vec{y} \mid \vec{x}, u_0)\right] &- \mathsf{KL}[\mathbb{P}_T(\,\mathrm{d}u_T \mid \vec{x}, \vec{y}) \parallel \mathbb{Q}^\theta(\,\mathrm{d}u_T)] \\ &- \mathbb{E}_{\mathbb{P}}\left[\mathsf{KL}[\mathbb{P}(\,\mathrm{d}u_{0:T-1} \mid \vec{x}, \vec{y}, u_T) \parallel \mathbb{Q}^\theta(\,\mathrm{d}u_{0:T} \mid u_T)]\right].\end{aligned} \tag{A.2.2}$$

Repeatedly applying the KL divergence chain rule to condition on $u_{T-1}, u_{T-2}, \ldots, u_1$ and using the Mavkov assumption yields

$$= \mathbb{E}_{\mathbb{P}}\left[\log q^\theta(y \mid x, u_0)\right] - \mathsf{KL}[\mathbb{P}_T(\,\mathrm{d}u_T \mid \vec{x}, \vec{y}) \parallel \mathbb{Q}^\theta(\,\mathrm{d}u_T)] \tag{A.2.3}$$

$$- \sum_{t=1}^{T} \mathbb{E}_{\mathbb{P}}\left[\mathsf{KL}[\mathbb{P}(\,\mathrm{d}u_{t-1} \mid u_t, \vec{x}, \vec{y}) \parallel \mathbb{Q}^\theta(\,\mathrm{d}u_{t-1} \mid u_t)]\right]. \tag{A.2.4}$$

□

A.2.2 Parametrization and Re-Weighting

By Equation (4.10), our loss function depends on terms of the form

$$L_{t-1} = \frac{1}{2\widetilde{\beta}_t}||C^{-1/2}(\widetilde{m}_t(u_t, u_0) - m_t^\theta(u_t))||_{\mathcal{F}}^2. \tag{A.2.5}$$

That is, our model must predict the mean function $\widetilde{m}_t(u_t, u_0)$ given (t, u_t). By Proposition (3) and Proposition (2),

$$\widetilde{m}_t(u_t, u_0) = \frac{\sqrt{\gamma_{t-1}}\beta_t}{1-\gamma_t}u_0 + \frac{\sqrt{1-\beta_t}(1-\gamma_{t-1})}{1-\gamma_t}u_t \tag{A.2.6}$$

$$u_0 = \frac{1}{\sqrt{\gamma_t}}\left(u_t - \sqrt{1-\gamma_t}\xi\right) \tag{A.2.7}$$

where $\xi \sim \mathcal{N}(0, C)$. Combining these two expressions, we see that

$$\widetilde{m}_t(u_t, u_0) = \frac{\sqrt{\gamma_{t-1}}\beta_t}{(1-\gamma_t)\sqrt{\gamma_t}}\left(u_t - \sqrt{1-\gamma_t}\right) + \frac{\sqrt{1-\beta_t}(1-\gamma_{t-1})}{1-\gamma_t}u_t \tag{A.2.8}$$

$$= \frac{1}{\sqrt{1-\beta_t}}\left(u_t - \frac{\beta_t}{\sqrt{1-\gamma_t}}\xi\right). \tag{A.2.9}$$

We thus parametrize the variational mean via

$$m_t^\theta(u_t) = \frac{1}{\sqrt{1-\beta_t}}\left(u_t - \frac{\beta_t}{\sqrt{1-\gamma_t}}\xi_t^\theta(u_t)\right). \tag{A.2.10}$$

Because C is a linear operator, $C^{-1/2}$ must also be a linear operator. Thus, plugging in our reparametrized expressions for $\widetilde{m}_t(u_t, u_0)$ and $m_t^\theta(u_t)$, we see that

$$L_{t-1} = \frac{\beta_t^2}{2\widetilde{\beta}_t(1-\beta_t)(1-\gamma_t)}||C^{-1/2}(\xi - \xi_t^\theta(u_t))||_{\mathcal{F}}^2. \tag{A.2.11}$$

We thus have obtained Equation (4.12).

A.3 Covariance Operators

Recall that for a Gaussian measure $F \sim \mathbb{P}_F = \mathcal{N}(m, C)$ on a separable Hilbert space $\mathcal{F}$, the covariance operator $C : \mathcal{F} \to \mathcal{F}$ is defined via

$$Cg = \int_{\mathcal{F}} \langle g, F\rangle F \, \mathrm{d}\mathbb{P}_F - \langle g, m\rangle m \qquad \forall g \in \mathcal{F}. \tag{A.3.1}$$

We now focus on the case where our Gaussian measure is specified by a Gaussian process with mean $m \in \mathcal{F}$ and kernel $k : \mathcal{X}^2 \to \mathbb{R}$. Note that $k(x, x') = \mathbb{E}[(F(x) - m(x))\,(F(x') - m(x'))]$ specifies the covariance at points $x, x' \in \mathcal{X}$.

A.3.1 Square-Integrable Case

Consider first $\mathcal{F} = L^2(\mathcal{X}, \mu)$. In this setting, we have that

$$[Cg](x) = \int_{\mathcal{X}} k(x, x')g(x') \, \mathrm{d}\mu(x'). \tag{A.3.2}$$

This can be derived from Equation (A.3.1) via

$$\begin{aligned}
[Cg](x) &= \int_{\mathcal{F}} \langle g, F\rangle_{L^2(\mathcal{X},\mu)} F(x) \, \mathrm{d}\mathbb{P}_F - \langle g, m\rangle_{L^2(\mathcal{X},\mu)} m(x) && \text{(A.3.3)}\\
&= \int_{\mathcal{F}} \left[\int_{\mathcal{X}} g(x')F(x') \, \mathrm{d}\mu(x')\right] F(x) \, \mathrm{d}\mathbb{P}_F - \langle g, m\rangle_{L^2(\mathcal{X},\mu)} m(x) && \text{(A.3.4)}\\
&= \int_{\mathcal{X}} g(x') \left[\int_{\mathcal{F}} F(x)F(x') \, \mathrm{d}\mathbb{P}_F\right] \mathrm{d}\mu(x') - \langle g, m\rangle_{L^2(\mathcal{X},\mu)} m(x) && \text{(A.3.5)}\\
&= \int_{\mathcal{X}} g(x') \left[k(x, x') + m(x)m(x')\right] \mathrm{d}\mu(x') - \int_{\mathcal{X}} g(x')m(x')m(x) \, \mathrm{d}\mu(x') && \text{(A.3.6)}\\
&= \int_{\mathcal{X}} g(x')k(x, x') \, \mathrm{d}\mu(x'). && \text{(A.3.7)}
\end{aligned}$$

where we apply Fubini's theorem in the third equality.

Proof of Proposition (5)**.**

Proof. Set $\vec{z} = \{z^{(1)}, \ldots, z^{(n)}\} \subset \mathcal{X}$. If suffices to check the case that $\vec{x} = x \cup \vec{z}$ is increased by a single point $x \in \mathcal{X}$.

Let $K_{\vec{z}\vec{z}} \in \mathbb{R}^{n\times n}$ be the covariance matrix corresponding to $\vec{z}$, and let $K_{\vec{x}\vec{x}} \in \mathbb{R}^{(n+1)\times(n+1)}$ be the covariance matrix corresponding to $\vec{x}$, i.e. in both cases the covariance matrix is given by evaluating the kernel k at all combinations of points in $\vec{z}$ or $\vec{x}$. Let $k_{\vec{z}}(x) \in \mathbb{R}^n$ be the covariance between the points of $\vec{z}$ and our new point x. Lastly, let $\Delta m(\vec{z}) \in \mathbb{R}^n$ be any vector and be $\Delta m(x) \in \mathbb{R}$ any scalar. We will write $\Delta m(\vec{x}) = [\Delta m(\vec{z}), \Delta m(x)]^T \in \mathbb{R}^{n+1}$ for the vector extending $\Delta m(\vec{z})$ by the single entry $\Delta m(x)$.

Then, we have that

$$K_{\vec{x}\vec{x}} = \begin{bmatrix} K_{\vec{z}\vec{z}} & k_{\vec{z}}(x) \\ k_{\vec{z}}(x)^T & k(x, x) \end{bmatrix}, \tag{A.3.8}$$

i.e. the extended covariance matrix corresponding to $\vec{x}$ can be written as a block matrix containing the covariance matrix for $\vec{z}$. Our goal is to show that

$$\Delta m(\vec{z})^T K_{\vec{z}\vec{z}}^{-1} \Delta m(\vec{z}) \leq \Delta m(\vec{x})^T K_{\vec{x}\vec{x}}^{-1} \Delta(\vec{x}). \tag{A.3.9}$$

Using the block matrix inversion formula (see e.g. Williams and Rasmussen (2006, Appendix A.3)), we may express $K_{\vec{x}\vec{x}}^{-1}$ as

$$K_{\vec{x}\vec{x}}^{-1} = \begin{bmatrix} K_{\vec{z}\vec{z}}^{-1} + K_{\vec{z}\vec{z}}^{-1} k_{\vec{z}}(x) M k_{\vec{z}}(x)^T K_{\vec{z}\vec{z}}^{-1} & -K_{\vec{z}\vec{z}}^{-1} k_{\vec{z}}(x) M \\ -M k_{\vec{z}}(x)^T K_{\vec{z}\vec{z}}^{-1} & M \end{bmatrix} \tag{A.3.10}$$

where

$$M = \left(k(x,x) - k_{\vec{z}}(x)^T K_{\vec{z}\vec{z}}^{-1} k_{\vec{z}}(x)\right)^{-1} \in \mathbb{R}. \tag{A.3.11}$$

Note that M is exactly the posterior variance at $x \in \mathcal{X}$ of a GP with covariance function k (Williams and Rasmussen, 2006, Eqn. 2.26). In particular, we must have $M \geq 0$.

We now proceed to directly compute the quadratic form on the right-hand side of Equation (A.3.9). We have:

$$\begin{aligned}
&[\Delta m(\vec{z}), \Delta m(x)]\, K_{\vec{x}\vec{x}}^{-1} \begin{bmatrix} \Delta m(\vec{z}) \\ \Delta m(x) \end{bmatrix} && \text{(A.3.12)} \\
&= \left\langle \begin{bmatrix} \Delta m(\vec{z})^T \left(K_{\vec{z}\vec{z}}^{-1} + K_{\vec{z}\vec{z}}^{-1} k_{\vec{z}}(x) M k_{\vec{z}}(x)^T K_{\vec{z}\vec{z}}^{-1}\right) - \Delta m(x) M k_{\vec{z}}(x)^T K_{\vec{z}\vec{z}}^{-1} \\ -\Delta m(\vec{z})^T K_{\vec{z}\vec{z}}^{-1} k_{\vec{z}}(x) M + \Delta m(x) M \end{bmatrix}, \begin{bmatrix} \Delta m(\vec{z}) \\ \Delta m(x) \end{bmatrix} \right\rangle && \text{(A.3.13)} \\
&= \Delta m(\vec{z})^T K_{\vec{z}\vec{z}}^{-1} \Delta m(\vec{z}) + \Delta m(\vec{z})^T K_{\vec{z}\vec{z}}^{-1} k_{\vec{z}}(x) M k_{\vec{z}}(x)^T K_{\vec{z}\vec{z}}^{-1} \Delta m(\vec{z}) && \text{(A.3.14)} \\
&\qquad - \Delta m(x) M k_{\vec{z}}(x)^T K_{\vec{z}\vec{z}}^{-1} \Delta m(\vec{z}) - \Delta m(\vec{z})^T K_{\vec{z}\vec{z}}^{-1} k_{\vec{z}}(x) M \Delta m(x) + \Delta m(x)^2 M && \text{(A.3.15)} \\
&= \Delta m(\vec{z})^T K_{\vec{z}\vec{z}}^{-1} \Delta m(\vec{z}) + M \left(\Delta m(\vec{z})^T K_{\vec{z}\vec{z}}^{-1} k_{\vec{z}}(x) - \Delta m(x)\right)^2. && \text{(A.3.16)}
\end{aligned}$$

We now plug Equation (A.3.16) back into Equation (A.3.9). Noting the first term in (A.3.16) is precisely the LHS of (A.3.9), we only need to check

$$0 \leq M \left(\Delta m(\vec{z})^T K_{\vec{z}\vec{z}}^{-1} k_{\vec{z}}(x) - \Delta m(x)\right)^2. \tag{A.3.17}$$

However, note that we already observed that $M \geq 0$, and the other term is the square of a scalar, whence it is positive. □

A.3.2 Sobolev Case

We now consider the first-order Sobolev space $\mathcal{F} = H^1(\mathcal{X}, \mu)$.

We claim that

$$[Cg](x) = \int_{\mathcal{X}} k(x,x') g(x') \,\mathrm{d}\mu(x') + \int_{\mathcal{X}} \partial_{x'} k(x,x') \partial_{x'} g(x') \,\mathrm{d}\mu(x') \tag{A.3.18}$$

where we use the shorthand

$$\partial_{x'} k(x,x') = \frac{\partial}{\partial x'} k(x,x') \quad \text{and} \quad \partial_{x'} g(x') = \frac{\partial}{\partial x'} g(x'). \tag{A.3.19}$$

Note that the mean element is not dependent on the inner product – it is merely an arbitrary element $m \in \mathcal{F}$. Now, from

Equation (A.3.1),

$$[Cg](x) = \int_{\mathcal{F}} \langle g, F\rangle_{H^1(\mathcal{X},\mu)} F(x)\,\mathrm{d}\mathbb{P}_F - \langle g, m\rangle_{H^1(\mathcal{X},\mu)} \tag{A.3.20}$$

$$= \int_{\mathcal{F}} \left[\langle g, F\rangle_{L^2(\mathcal{X},\mu)} + \langle \partial_{x'} g(x'), \partial_{x'} F(x')\rangle_{L^2(\mathcal{X},\mu)}\right] F(x)\,\mathrm{d}\mathbb{P}_F - \langle g, m\rangle_{H^1(\mathcal{X},\mu)} m(x) \tag{A.3.21}$$

$$= \int_{\mathcal{X}} k(x,x') g(x')\,\mathrm{d}\mu(x') + \int_{\mathcal{F}} \langle \partial_{x'} g(x'), \partial_{x'} F(x')\rangle_{L^2(\mathcal{X},\mu)} F(x)\,\mathrm{d}\mathbb{P}_F - \langle \partial_{x'} g(x'), \partial_{x'} m(x')\rangle_{L^2(\mathcal{X},\mu)} m(x) \tag{A.3.22}$$

$$= \int_{\mathcal{X}} k(x,x') g(x')\,\mathrm{d}\mu(x') + \int_{\mathcal{X}} \partial_{x'} g(x') \mathbb{E}[F(x)\partial_{x'} F(x')]\,\mathrm{d}\mu(x') - \langle \partial_{x'} g(x'), \partial_{x'} m(x')\rangle_{L^2(\mathcal{X},\mu)} m(x) \tag{A.3.23}$$

$$= \int_{\mathcal{X}} k(x,x') g(x')\,\mathrm{d}\mu(x') + \int_{\mathcal{X}} \partial_{x'} g(x') \left(\partial_{x'} k(x,x') + m(x)\partial_{x'} m(x')\right)\,\mathrm{d}\mu(x') \tag{A.3.24}$$

$$- \langle \partial_{x'} g(x'), \partial_{x'} m(x')\rangle_{L^2(\mathcal{X},\mu)} m(x)$$

$$= \int_{\mathcal{X}} k(x,x') g(x')\,\mathrm{d}\mu(x') + \int_{\mathcal{X}} \partial_{x'} k(x,x') \partial_{x'} g(x,x')\,\mathrm{d}\mu(x'). \tag{A.3.25}$$

The third equality follows from the corresponding $L^2(\mathcal{X},\mu)$ calculation. The fifth equality follows from the fact that if $F \sim \mathcal{GP}(m,k)$ is differentiable with probability one, then $\partial_{x'} F$ is also a Gaussian process with mean $\partial_{x'} m$ (Williams and Rasmussen, 2006; Papoulis and Pillai, 2002), and moreover the covariance between F and its derivative is given by differentiating the kernel:

$$\mathrm{Cov}(F(x), \partial_{x'} F(x')) = \mathbb{E}\left[(F(x) - m(x))\left(\partial_{x'} F(x') - \partial_{x'} m(x')\right)\right] = \partial_{x'} k(x,x'). \tag{A.3.26}$$

See e.g. Williams and Rasmussen (2006, Chapter 9.4).

PSD Projection Details In Sobolev space, our discrete approximation to the KL divergence is given in terms of a quadratic form (see Section 5)

$$\mathsf{KL}[\mathcal{N}(m_1, C) \parallel \mathcal{N}(m_2, C)] \approx \Delta m(\vec{x})^T \left[I + D^T D\right] \left[K_{\vec{x}\vec{x}} + K'_{\vec{x}\vec{x}} D\right]^{-1} \Delta m(\vec{x}). \tag{A.3.27}$$

In practice, we parametrize D by a first-order difference operator when $\mathcal{X} = [0,1]$. For example, one choice of D can be constructed by using the forward difference equation at the left boundary, the backward difference equation at the right boundary, and the central difference equation on the interior of $[0,1]$.

Although the covariance operator C is symmetric and positive semi-definite (*with respect to the $H^1(\mathcal{X},\mu)$ inner product*) in theory, upon discretization we often obtain a non-PSD quadratic form. Thus, when naively used as a loss function, this quadratic form is unbounded from below, which leads to divergent training. We overcome this by projecting

$$A = \left[I + D^T D\right] \left[K_{\vec{x}\vec{x}} + K'_{\vec{x}\vec{x}} D\right]^{-1} \tag{A.3.28}$$

to a symmetric PSD matrix. In particular, we apply the methods of Cheng and Higham (1998); Higham (1988) to find

$$\widetilde{A} = \arg\min_B \{||B - A||_F : B \text{ is symmetric, PSD}\} \tag{A.3.29}$$

i.e. the closest symmetric PSD matrix to A in terms of the Frobenius norm. This has a unique solution, which can be computed in a straightforward manner. We briefly review this method here for the sake of completeness. First, set

$$C = \frac{1}{2}(A + A^T) \tag{A.3.30}$$

to be the symmetric part of A. Then, compute the usual spectral decomposition

$$C = Q\mathrm{diag}(\lambda_i) Q^T \tag{A.3.31}$$

where Q is a matrix containing the eigenvectors of C with corresponding eigenvalues $\{\lambda_i\}$. Set $\tau_i = \max(0, \lambda_i)$. Then,

$$\widetilde{A} = Q\mathrm{diag}(\tau_i) Q^T. \tag{A.3.32}$$

We will write $\widetilde{A} = \pi_{\mathrm{PSD}}(A)$ for this projection.

A.4 Model Details

In all of our experiments, our model architecture is the Graph Neural Operator (GNO) of Li et al. (2020). We use a width of 64, a kernel width of 256, and a depth of 6. Inputs to the GNO are graphs, constructed from discrete functional observations. In particular, for every function we construct a graph where each node corresponds to a single observation of the function. Each node has features corresponding to the observation location (i.e. point in $\mathcal{X}$), function value (i.e. scalar in $\mathbb{R}$), and additionally time step $t \in [1, T]$. Nodes are connected if the Euclidean distance between their observation locations is smaller than a fixed radius r. We use $r = 0.5$ in all of our experiments, and we additionally scale $\mathcal{X}$ to $[0, 1] \subset \mathbb{R}$. Each edge in our graph has features corresponding to the observation locations and function values of the respective nodes. While using $r = 1$ would be ideal in this setting, we find this to be prohibitively expensive in terms of computation and memory usage. The Fourier Neural Operator (FNO) (Li et al., 2021) has a significantly reduced computation and memory cost compared to the GNO, but this model is limited to functional observations which are on a uniform gridding of $\mathcal{X}$.

Our models are all trained for 50 epochs and a learning rate of 0.001.

We use $T = 1000$ time steps in all of our experiments. We set $\beta_1 = 10^{-4}$ and $\beta_T = 0.02$, and we linearly interpolate between these two values for other settings of β_t. We parametrize the Gaussian measure in our forward process via a mean-zero Gaussian process with a Matérn kernel of unit variance and lengthscale $\ell = 0.1$. In particular, we use a Matérn kernel with $\nu = 1/2$ (i.e. the exponential kernel) when $\mathcal{F} = L^2(\mathcal{X}, \mu)$ and $\nu = 3/2$ when $\mathcal{F} = H^1(\mathcal{X}, \mu)$. This choice was made to ensure that the Gaussian measure was sufficiently rough to remove any information contained in the functional data, yet regular enough to be square-integrable (and differentiable in the $\nu = 3/2$ case) such that we obtain a valid Gaussian measure on $\mathcal{F}$.

A.4.1 Kernel Ablation

In Tables 2-3, we study the effect of the kernel choice in the forward process on the MoGP and AEMET datasets. In particular, we train models as above (using the discrete $L^2(\mathcal{X}, \mu)$ loss function), but choose between values of $\nu = 1/2$ and $\nu = 3/2$ and sweep across various length scales between 0.005 and 0.5. We then sample 500 generated functions from our model, and compute the average pointwise mean and variance curves, as well as the average autocorrelation curve – see Figures (1) and (4) for a visualization. We report the MSE between these generated functional statistics and the true functional statistics given by the training data.

We see that choosing a length scale that is either significantly larger or smaller than the length scale of the underlying functional data can have negative effects on the statistics, but for reasonable choices of the length scale, the statistics are comparable. Although $\ell = 0.1$ does not produce the best MSE values on the AEMET datsaet, we still use $\ell = 0.1$ in our main experiments as this produced the most qualitatively realistic generated curves.

Table 2: Effect of kernel choice on the MoGP dataset. We report the MSE between various functional statistics on the training data and data generated via our model with the listed kernel hyperparameters.

ν	ℓ	Mean	Var.	Autocorr.
1/2	0.005	3.0333	6.1184	1.211e-4
	0.01	0.4474	1.2174	5.552e-06
	0.1	0.0032	0.2328	9.169e-06
	0.2	0.4496	1.2752	9.183e-06
	0.5	0.0318	0.2772	1.080e-05
3/2	0.005	0.5225	0.5783	3.638e-05
	0.01	1.6557	4.7887	5.699e-05
	0.1	0.4645	0.1239	1.928e-05
	0.2	0.1046	0.2300	3.947e-06
	0.5	0.2651	0.2586	6.677e-05

Table 3: Effect of kernel choice on the AEMET dataset. We report the MSE between various functional statistics on the training data and data generated via our model with the listed kernel hyperparameters. For $\nu = 3/2$ with a length scale of $\ell = 0.5$ our training failed to produce a reasonable model.

ν	ℓ	Mean	Var	Autocorr
1/2	0.005	0.1118	74.8143	1.813-06
	0.01	0.0646	2.2001	4.563e-06
	0.1	0.7284	2.2519	5.805e-05
	0.2	0.0152	1.0748	2.551e-06
	0.5	0.0832	3.0590	1.516e-05
3/2	0.005	0.1393	8.5001	1.700e-05
	0.01	0.0899	1.28130	5.638-06
	0.1	0.9317	148.4542	0.0021
	0.2	7.3748	15634.6889	0.0398
	0.5	-	-	-

A.5 Pseudocode

In this section we detail pseudocode for model training, unconditional sampling, and conditional sampling. Note that during training, we assume $u_0(\vec{x}) = \vec{y}$, i.e. we treat the observations as if they were noiseless. Thus the likelihood term $q(\vec{y} \mid \vec{x}, u_0)$ does not contribute to the loss, and we need only optimize the terms L_{t-1} (see Equation (4.12). Moreover, as mentioned in the main paper, we set $\lambda_t = 1$ as is standard in diffusion modeling (Ho et al., 2020).

Note that the given pseudocode for conditional generation covers both hard and soft conditioning. Hard conditioning is obtained when $n_{\text{free}} = 0$, and soft conditioning is obtained by setting $n_{\text{free}} \geq 1$, i.e. the parameter n_{free} indicates how many generation steps are not conditioned on the given information.

Algorithm 1: Training Step

Sample $(\vec{x}, \vec{y})$ from training data;
Sample t uniformly from $\{2, \dots, T\}$;
Sample $\xi \sim \mathcal{GP}(0, k)$, evaluated at $\vec{x}$ to obtain $\xi(\vec{x})$;
Construct $u_t \mid u_0$, evaluated at $\vec{x}$, via Lemma (2): $u_t(\vec{x}) = \sqrt{\gamma_t} u_0(\vec{x}) + \sqrt{1-\gamma_t}\xi(\vec{x})$;
Compute model output $\xi^\theta(\vec{x} \mid u_t, t)$;
Take a θ-gradient step on $L_{t-1} = \big(\xi(\vec{x}) - \xi^\theta(\vec{x} \mid u_t, t)\big)^T A \big(\xi(\vec{x}) - \xi^\theta(\vec{x} \mid u_t, t)\big)$, where

$$A = \begin{cases} K_{\vec{x}\vec{x}}^{-1} & \mathcal{F} = L^2(\mathcal{X}, \mu) \\ \pi_{\text{PSD}}\left([I + D^T D][K_{\vec{x}\vec{x}} + K'_{\vec{x}\vec{x}} D]^{-1}\right) & \mathcal{F} = H^1(\mathcal{X}, \mu) \end{cases} \tag{A.5.1}$$

Algorithm 2: Unconditional Sampling

Specify query points $\vec{x} \subset \mathcal{X}$;
Sample $u_T \sim \mathcal{GP}(0, k)$, evaluated at $\vec{x}$ to obtain $u_T(\vec{x})$;
for $t = T, T-1, \dots, 1$ **do**
    Sample $\xi_t \sim \mathcal{GP}(0, k)$, evaluated at $\vec{x}$ to obtain $\xi_t(\vec{x})$;
    $u_{t-1}(\vec{x}) \leftarrow \frac{1}{\sqrt{1-\beta_t}}\left(u_t(\vec{x}) - \frac{\beta_t}{\sqrt{1-\gamma_t}}\xi^\theta(\vec{x} \mid u_t, t)\right) + \sqrt{\widetilde{\beta}_t}\xi_t(\vec{x})$;
end for
Return $u_0(\vec{x})$

Algorithm 3: Conditional Sampling

Given: conditioning information $\mathcal{D} = \{(x_c^{(i)}, y_c^{(i)})\}_{i=1}^{n_c} = \{\vec{x}_c, \vec{y}_c\}$;

Specify query points $\vec{x} = \{x^{(1)}, x^{(2)}, \ldots, x^{(n)}\} \subset \mathcal{X}$;

Create augmented support $\vec{z} = \{x^{(1)}, x^{(2)}, \ldots, x^{(n)}, x_c^{(1)}, \ldots, x_c^{(n_c)}\}$;

Sample $u_T \sim \mathcal{GP}(0, k)$, evaluated at $\vec{z}$ to obtain $u_T(\vec{z})$;

for $t = T, T-1, \ldots, 1$ **do**

  Sample $\xi_t \sim \mathcal{GP}(0, k)$, evaluated at $\vec{z}$ and $\vec{x}_c$ to obtain $\xi_t(\vec{z})$;

  Sample reverse process unconditionally on $\vec{z}$:

$$\widetilde{u}_{t-1}(\vec{z}) \leftarrow \frac{1}{\sqrt{1-\beta_t}}\left(u_t(\vec{z}) - \frac{\beta_t}{\sqrt{1-\gamma_t}}\xi^\theta(\vec{z} \mid u_t, t)\right) + \sqrt{\widetilde{\beta}_t}\xi_t(\vec{z}) \tag{A.5.2}$$

  if $t > n_{free}$ **then**

    Sample $\xi_t' \sim \mathcal{GP}(0, k)$, and evaluate at $\vec{x}_c$ to obtain $\xi_t'(\vec{x}_c)$;

    Perturb conditioning information via the forward process:

$$\vec{y}_{c,t} = \sqrt{\gamma_t}\vec{y}_c + \sqrt{1-\gamma_t}\xi_t'(\vec{x_c}) \tag{A.5.3}$$

    For each $x \in \vec{z}$, conditioned on perturbed conditioning information by setting

$$u_{t-1}(x) = \begin{cases} \widetilde{u}_{t-1}(x) & x \notin \mathcal{D} \\ y_{c,t}(x) & x \in \mathcal{D} \end{cases} \tag{A.5.4}$$

  else

    Do no conditioning: $u_{t-1}(\vec{z}) \leftarrow \widetilde{u}_{t-1}(\vec{z})$;

end for

Return u_0

A.6 Additional Experiments

A.6.1 Unconditional Samples

In Figure (4), we provide additional examples of our model on various datasets not discussed in the main paper. The first dataset (*Linear*) is a synthetic dataset consisting of random linear functions $u_0(x) = ax + b$ where $a \sim \mathcal{N}(2, 0.25^2)$ and $b \sim \mathcal{N}(-1, 0.07^2)$. Note that, although the pointwise variance of the generated samples in this dataset appear to be significantly smaller than the that of the true samples, this is largely due to the small scale of the variance. The other datasets (*Growth, Canadian, Octane*) are well-known functional data analysis datasets, which are available in the Python package scikit-fda (Ramos-Carreño et al., 2019).

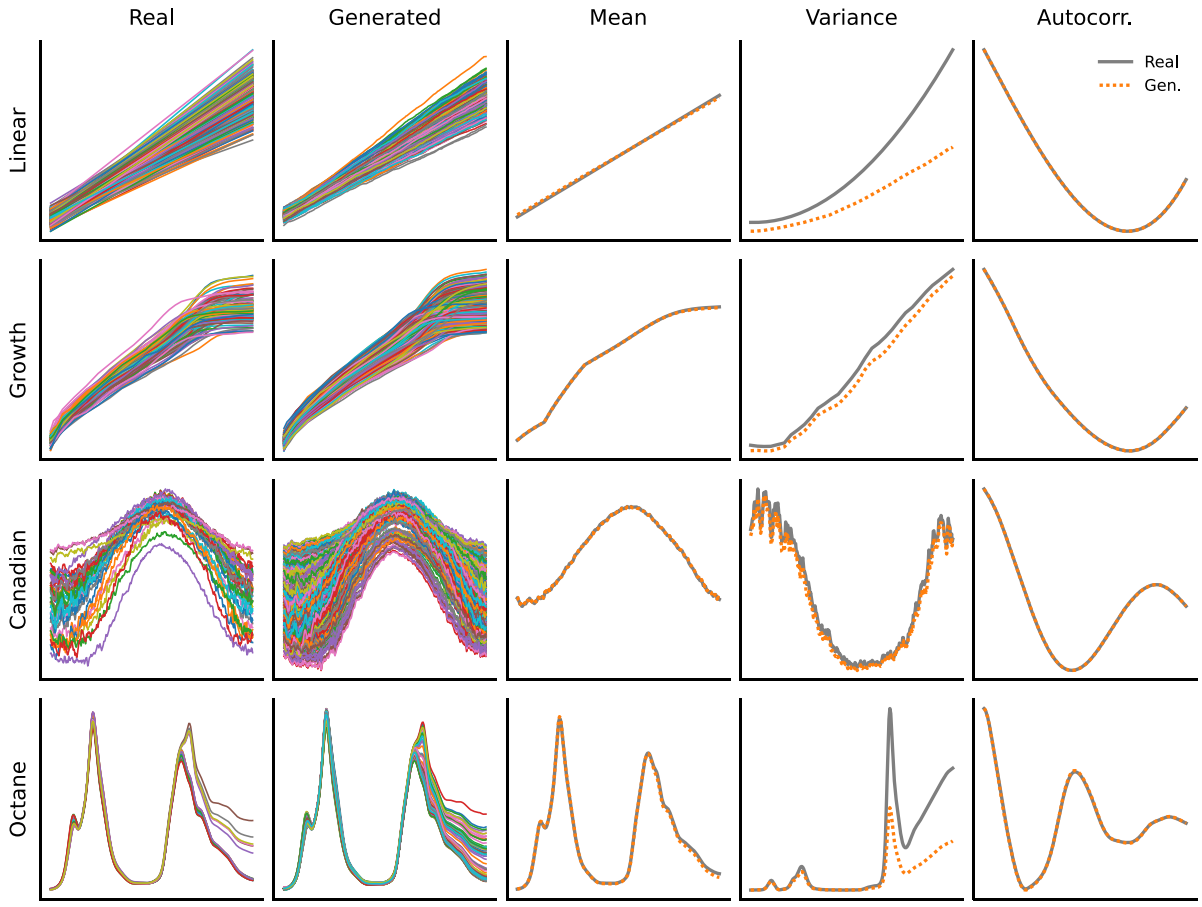

Figure 4: Unconditional function generation on a synthetic (Linear) and several real-world (Growth, Canadian, Octane) datasets. For each dataset, a GNO model was trained on the plotted functions (first column), and a total of 500 functions were sampled from the model (second column).

A.6.2 FPCA Baseline

We additionally include a simple unconditional baseline based on functional principal component analysis (FPCA). In particular, we approximate the first $M = 5$ functional principal components by discretizing the training data (see Ramsay and Silverman (2008, Chapter 6) for details and Ramos-Carreño et al. (2019) for an implementation), followed by fitting a multivariate Gaussian to the resulting scores. To sample from this model, we sample from the Gaussian distribution over scores and project back to function space by taking linear combinations of the principal components with these sampled scores.

See Figure 5 for an illustration of this approach on all of the datasets we have thus far considered. We see that while the

FPCA baseline is able to accurately match the functional statistics of the training data, the generated samples often fail to match the qualitative performance of our FuncDiff model (Figures 1 and 4). Note that, unlike our FuncDiff model, we are unable to perform conditional generation with this FPCA baseline.

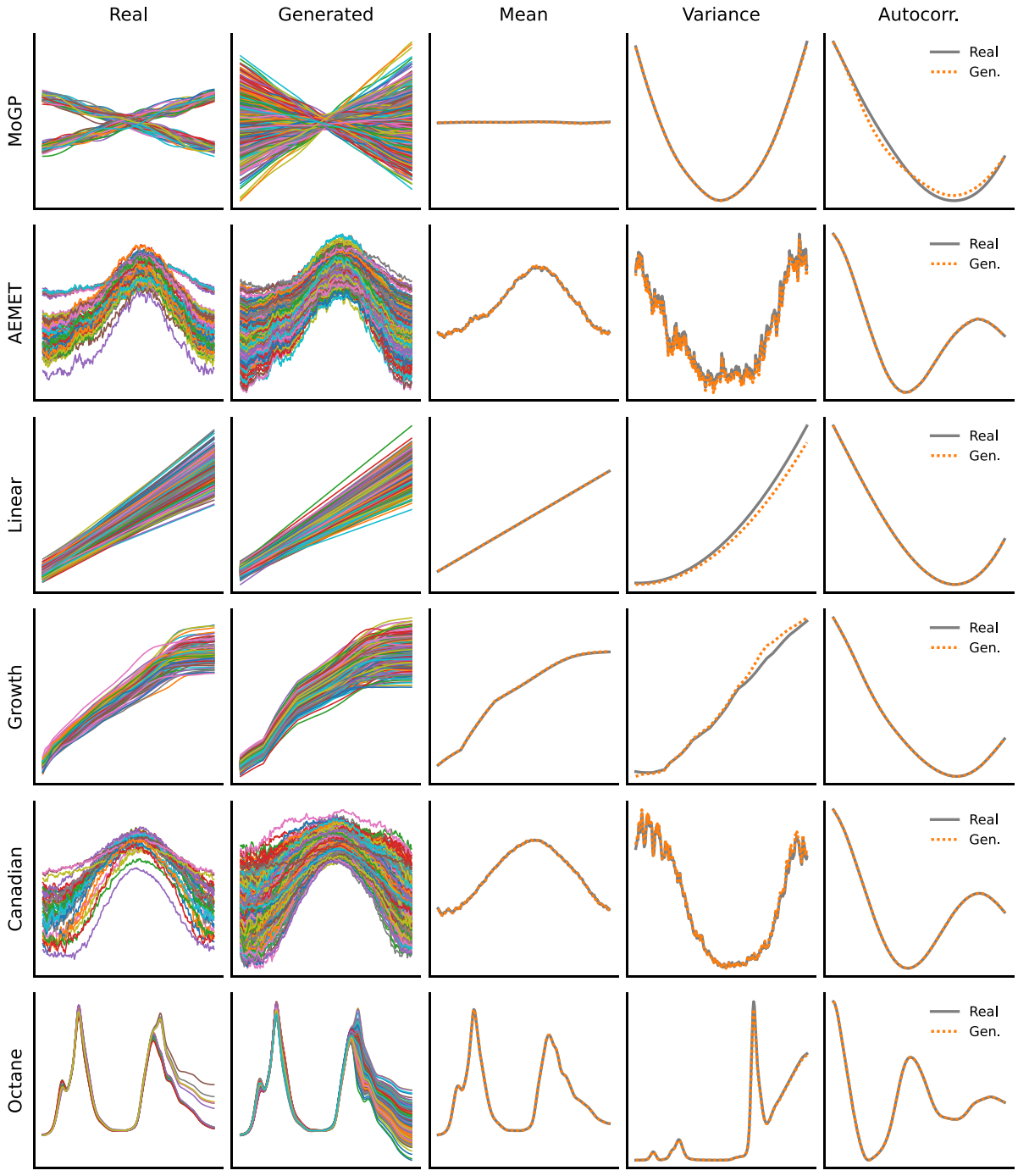

Figure 5: Unconditional samples from an FPCA-based model on various datasets. For each dataset, we estimate the first $M = 5$ functional principal components and fit a Gaussian distribution to the resulting scores. Generation is performed by sampling from said Gaussian and taking the resulting linear combination of functional principal components. Although the functional statistics closely match those of the training data, the perceptual quality of the generated curves is worse than our FuncDiff model.

A.6.3 Spectral Loss

In the main paper, we approximate the functional KL divergence by discretizing the underlying operators. In this section, we experiment with an alternative approach based on the spectrum of the covariance operator. We focus here on the setting $\mathcal{F} = L^2(\mathcal{X}, \mu)$ with $\mathcal{X} = [0, 1]$ equipped with the Lebesgue measure $\mu = \mathrm{d}x$. Consider a Gaussian measure on $\mathcal{F}$ with covariance operator C. Since C is self-adjoint and compact, the spectral theorem tells us that the eigenfunctions of C form an orthonormal basis of $\mathcal{F}$. We denote the eigenvalues and eigenfunctions of C by $\{(\lambda_j, e_j)\}_{j=1}^{\infty}$. We then have that (Da Prato and Zabczyk, 2014, Remark 2.24)

$$\mathsf{KL}[\mathcal{N}(m_1, C) \parallel \mathcal{N}(m_2, C)] = \frac{1}{2}\langle m_1 - m_2, C^{-1}(m_1 - m_2)\rangle_{L^2(\mathcal{X},\mu)} \tag{A.6.1}$$

$$= \frac{1}{2}\sum_{j=1}^{\infty} \lambda_j^{-1}\langle m_1 - m_2, e_j\rangle^2_{L^2(\mathcal{X},\mu)} \approx \frac{1}{2}\sum_{j=1}^{J} \lambda_j^{-1}\langle m_1 - m_2, e_j\rangle^2_{L^2(\mathcal{X},\mu)}. \tag{A.6.2}$$

Thus, an alternative method for approximating the KL divergence between Gaussian measures with equal covariance operators is to truncate the above sum at some specified number of terms J. For some choices of C, the eigenvalues and eigenfunctions are analytically known – for example, see Williams and Rasmussen (2006, Chapter 4) for the squared-exponential kernel, and see Le Maître and Knio (2010, Chapter 2) or Burt (2018, Section 2.5) for the exponential kernel.

In Figure 6, we compare this spectral approach to the discrete approach proposed in Section (5). In particular, we specify C via a Gaussian process with a Matérn kernel with $\nu = 1/2$, unit variance, and lengthscale $\ell = 0.1$. This is done to match the settings in our other experiments. Moreover, the eigenvalues and eigenfunctions are analytically available in this case (Le Maître and Knio, 2010; Burt, 2018). In each row of Figure 6, we specify particular functions for m_1 and m_2. We vary the discretization size (i.e. the number of function observations) on the horizontal axis for discretization sizes of $10, 50, 100, 300$, and plot the estimated KL divergence between $\mathcal{N}(m_1, C)$ and $\mathcal{N}(m_2, C)$ on the vertical axis.

We observe that the discrete approximation to the KL divergence (in blue) is monotonically increasing, as was proved in Proposition (5). However, we see that the spectral approximation is sensitive to both the number of terms in the series expansion and the discretization size. In particular, when using $J = 10$ terms, the spectral approximation underestimates the true KL divergence. In contrast, when $J \geq 50$, we see that the spectral approximation overestimates the true KL divergence by several orders of magnitude if the discretization of $\mathcal{X}$ is not sufficiently fine. This effect worsens as we increase the number of terms J. We conjecture that this is because the eigenfunctions e_j are sinusoidal in this case, and thus without a sufficiently fine discretization of $\mathcal{X}$, the inner product in the spectral approximation is a poor numerical estimate of the true inner product.

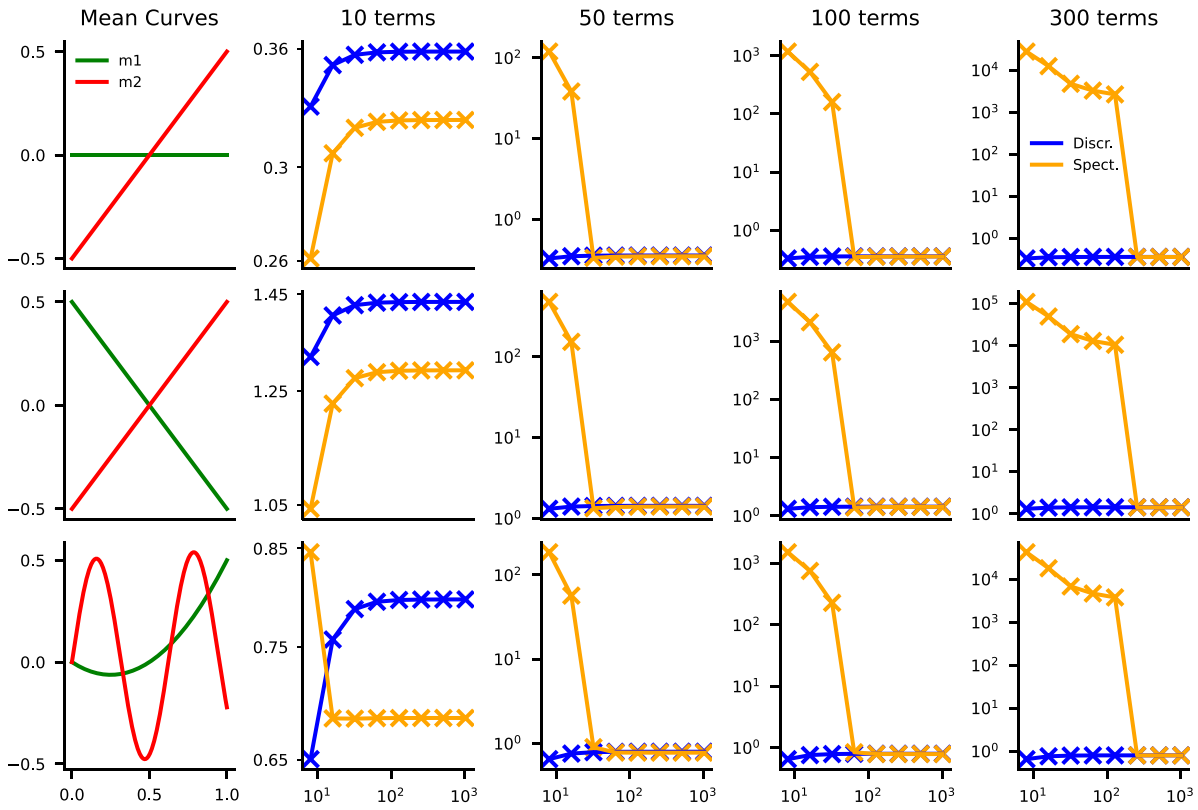

Figure 6: Various synthetic functions (first column) and estimates of the KL divergence between Gaussian measures with these means, having covariance operator given by an exponential kernel. For columns 2-5, the horizontal axis corresponds to discretization size (i.e. number of function observations), and the vertical axis corresponds to the corresponding estimated KL divergence. The discrete method (in blue) has KL estimates that are monotonically increasing (see also Proposition (5)), but the spectral method (in orange) is sensitive to the choice of terms in the series expansion as well as the discretization size.