
Functional Flow Matching

Gavin Kerrigan

Department of Computer Science
University of California, Irvine
gavin.k@uci.edu

Giosue Migliorini

Department of Statistics
University of California, Irvine
gmiglior@uci.edu

Padhraic Smyth

Department of Computer Science
University of California, Irvine
smyth@ics.uci.edu

Abstract

We propose Functional Flow Matching (FFM), a function space generative model that generalizes the recently introduced Flow Matching model to operate in infinite-dimensional spaces. Our approach works by first defining a path of probability measures that interpolates between a fixed Gaussian measure and the data distribution, followed by learning a vector field on the underlying space of functions that generates this path of measures. Our method does not rely on likelihoods or simulations, making it well-suited to the function space setting. We provide both a theoretical framework for building such models and an empirical evaluation of our techniques. We demonstrate through experiments on several real-world benchmarks that our proposed FFM method outperforms several recently proposed function space generative models.

1 Introduction

Generative models have seen a meteoric rise in capabilities on various domains, such as images [Dhariwal and Nichol, 2021, Kang et al., 2023], video [Saharia et al., 2022, Ho et al., 2022], and audio [Kong et al., 2021, Goel et al., 2022]. Despite these successes, many methods implicitly assume that the distribution of interest is supported on a finite-dimensional space. However, there are multiple important applications that involve data that is inherently infinite-dimensional. For instance, draws from a time series, solutions to partial differential equations, and audio signals are naturally represented as *functions*.

For functional data, typical generative models operate directly on a discretization of the data [Tashiro et al., 2021, Rasul et al., 2021a, Yan et al., 2021]. However, these approaches are often tied to a chosen discretization and are ill-posed in the functional limit. To overcome these limitations, recent methods have proposed building generative models directly in function spaces. For instance, Kerrigan et al. [2023] and Lim et al. [2023a] propose function space diffusion models, and Rahman et al. [2022] propose a functional GAN.

We add to this growing literature on function space generative models. In particular, we propose Functional Flow Matching (FFM), a continuous-time normalizing flow model for functional data. Given that Euclidean normalizing flow methods [Papamakarios et al., 2021, Kobyzev et al., 2020] are posed in terms of densities, which generally do not exist in infinite-dimensional spaces, a key challenge of performing this generalization is to pose a purely measure-theoretic model. In particular, our model is a generalization of the recently proposed Flow Matching model of Lipman et al. [2023].

Our proposed FFM model first constructs a path of conditional Gaussian measures, approximately interpolating between a fixed reference Gaussian measure and a given function. A path of measures interpolating between said reference measure and the data distribution is then obtained by marginalizing these conditional paths over the data distribution. We learn a vector field on our space of functions which approximately generates this path of measures, allowing us to generate samples from our data distribution by solving a differential equation. Figure 1 illustrates our approach.

Our approach allows for simulation-free training, in the sense that no samples are drawn from the model. Moreover, our training objective is regression based, allowing us to avoid pathologies with regards to maximum likelihood training in a functional setting. We empirically verify our framework on several time series datasets and a fluid dynamics dataset, demonstrating that the FFM model outperforms several competitive function space models across a variety of domains.

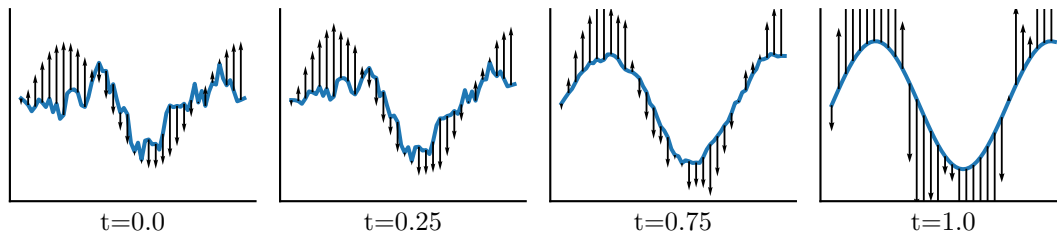


Figure 1: An illustration of our FFM method. The vector field $v_t(f) \in \mathcal{F}$ (in black) transforms a noise sample $g \sim \mu_0 = \mathcal{N}(0, C_0)$ drawn from a Gaussian process with a Matérn kernel (at $t = 0$) to the function $f(x) = \sin(x)$ (at $t = 1$) via solving a function space ODE. By sampling many such $g \sim \mu_0$, we define a conditional path of measures μ_t^f approximately interpolating between $\mathcal{N}(0, C_0)$ and the function f , which we marginalize over samples $f \sim \nu$ from the data distribution in order to obtain a path of measures approximately interpolating between μ_0 and ν .

2 Related Work

Flow Matching and Normalizing Flows We generalize the Flow Matching model of Lipman et al. [2023], which is a novel approach to simulation-free continuous-time normalizing flows [Chen et al., 2018, Papamakarios et al., 2021, Kobyzev et al., 2020]. This approach has demonstrated impressive capabilities on several image generation tasks. However, Flow Matching and other recently proposed simulation-free continuous normalizing flows have only been explored for data distributions supported on finite-dimensional spaces, such as Euclidean spaces [Lipman et al., 2023, Albergo and Vanden-Eijnden, 2023, Liu et al., 2022, Neklyudov et al., 2022] and Riemannian manifolds [Chen and Lipman, 2023, Ben-Hamu et al., 2022]. In contrast, we propose Functional Flow Matching, a continuous-time normalizing flow for infinite-dimensional data. To the best of our knowledge, this is the first normalizing flow model posed in infinite-dimensional spaces.

Function Space Generative Models Recently, a number of authors have proposed function space generalizations of various deep generative models. Close in spirit to our work are those generalizing diffusion models [Ho et al., 2020, Song et al., 2021, Song and Ermon, 2019] to the infinite-dimensional setting. In particular, Kerrigan et al. [2023] and Lim et al. [2023a] propose function space generalizations of discrete-time diffusion models, whereas Pidstrigach et al. [2023], Franzese et al. [2023] and Hagemann et al. [2023] propose function space generalizations of continuous-time diffusion models. Beyond diffusion models, function space GANs [Rahman et al., 2022] and energy-based models [Lim et al., 2023b] have also been proposed. Our work adds to this growing literature on function space generative models by proposing the first function space normalizing flow model.

Discrete Functional Generative Models While there has been growing interest in developing generative models directly in infinite-dimensional spaces, there has also been work proposing generative models for functional data that operate directly on a discretization of the underlying space, for example, diffusion models for time series [Tashiro et al., 2021, Rasul et al., 2021a, Yan et al., 2021] (see Lin et al. [2023] for a recent survey on these methods). Other models, such as normalizing flows [Rasul et al., 2021b], latent variable models [Zhou et al., 2022, Rubanova et al., 2019, Yildiz et al., 2019], and GANs [Yoon et al., 2019, Kidger et al., 2021] have also been explored. However, these methods all operate directly on the discrete observations of a given time series. This has several drawbacks: for instance, it is difficult to transfer a model trained on one discretization to another, and often these models are ill-posed in the functional limit (i.e. as the discretization size goes to zero). In contrast, our work begins from a function space point of view, where we only discretize in order to perform computations.

3 Notation and Background

We begin by introducing some notation and background which we will later use to construct our model. Section 3.1 introduces notions related to flows on function spaces, and Section 3.2 introduces the weak continuity PDE [Stepanov and Trevisan, 2017] which plays a key role in our constructions.

3.1 Preliminaries

Let $\mathcal{X} \subset \mathbb{R}^d$ and consider a real separable Hilbert space \mathcal{F} of functions $f : \mathcal{X} \rightarrow \mathbb{R}$ equipped with the Borel σ -algebra $\mathcal{B}(\mathcal{F})$. We consider the setting where there is a probability measure ν on \mathcal{F} from which we have samples, i.e. random functions drawn from the

data distribution ν . Our goal is to build a generative model which allows us to sample from ν . Importantly, any such generative model should be discretization-invariant, in the sense that the model should be able to generate functions which may be observed on any finite but arbitrary discretization of \mathcal{X} .

In this work, we consider paths of probability measures $(\mu_t)_{t \in [0,1]}$, where $\mu_t \in \mathcal{P}(\mathcal{F})$ is a probability measure on \mathcal{F} at every time $t \in [0,1]$. In particular, we will construct a path of measures which approximately interpolates between a fixed reference measure μ_0 at time $t = 0$ and the data distribution at time $t = 1$, so that $\mu_1 \approx \nu$. This interpolation is approximate in the sense that μ_1 will be a smoothed version of the data distribution, obtained from ν via convolution with a Gaussian measure having small variance [Bogachev, 1998, Appendix A].

We consider paths of probability measures which are generated locally, in the sense that there is some underlying time-dependent vector field on \mathcal{F} such that the path of measures $(\mu_t)_{t \in [0,1]}$ is obtained by flowing samples $g \sim \mu_0$ along said vector field. More formally, a (*time-dependent*) *vector field* on \mathcal{F} is a mapping $v : [0,1] \times \mathcal{F} \rightarrow \mathcal{F}$.

The *flow* associated to a vector field $(v_t)_{t \in [0,1]}$ is the mapping $\phi : [0,1] \times \mathcal{F} \rightarrow \mathcal{F}$ specified by the initial value problem

$$\partial_t \phi_t(g) = v_t(\phi_t(g)) \quad \phi_0(g) = g. \quad (1)$$

As written, Equation (1) represents an ordinary differential equation (ODE) on the abstract and potentially infinite-dimensional space \mathcal{F} . Such ODEs are often dubbed *abstract differential equations* [O'Regan, 1997, Zaidman, 1999]. We assume that all vector fields in this work are sufficiently regular such that a solution to Equation (1) is guaranteed to exist for all $t \in [0,1]$ and ν -a.e. initial condition g .

Given any initial probability measure $\mu_0 \in \mathcal{P}(\mathcal{F})$, we may consider the path of probability measures generated by the flow ϕ . That is, for any $t \in [0,1]$ we define the measure μ_t via the pushforward of μ_0 along ϕ_t , i.e. $\mu_t = [\phi_t]_{\#} \mu_0$, so that $\mu_t(A) = \mu_0(\phi_t^{-1}(A))$ for any measurable $A \subset \mathcal{F}$. Here, ϕ_t is assumed to be measurable for all $t \in [0,1]$.

3.2 Weak Continuity PDE

Previously, we noted how one may obtain a path of probability measures from an initial probability measure $\mu_0 \in \mathcal{P}(\mathcal{F})$ by considering the pushforward of μ_0 along the flow of a given vector field $(v_t)_{t \in [0,1]}$.

Conversely, we say that the vector field $(v_t)_{t \in [0,1]}$ *generates* the path of measures $(\mu_t)_{t \in [0,1]}$ if the path

$(\mu_t)_{t \in [0,1]}$ is obtained via the pushforward of μ_0 along the flow associated with $(v_t)_{t \in [0,1]}$. Directly verifying whether a vector field generates a given path of measures (by verifying the pushforward relationship) is typically infeasible. Instead, we can check if the two satisfy the *continuity equation*

$$\partial_t \mu_t + \operatorname{div}(v_t \mu_t) = 0 \quad \text{on } \mathcal{F} \times [0,1]. \quad (2)$$

We interpret this partial differential equation (PDE) in the weak sense [Ambrosio et al., 2005, Ch. 8], by which we mean that the pair $(v_t)_{t \in [0,1]}$ and $(\mu_t)_{t \in [0,1]}$ satisfy Equation (2) if

$$\int_0^1 \int_{\mathcal{F}} (\partial_t \varphi(g, t) + \langle v_t(g), \nabla_g \varphi(g, t) \rangle) d\mu_t(g) dt = 0 \quad (3)$$

for all $\varphi : \mathcal{F} \times [0,1] \rightarrow \mathbb{R}$ in an appropriate space of test functions. Typically, φ is assumed to be *cylindrical*, i.e. of the form $\varphi(f, t) = \psi(\pi(f), t)$ where $\pi : g \mapsto (\langle g, e_1 \rangle, \dots, \langle g, e_d \rangle)$ is a d -dimensional projection of g via an orthonormal system $(e_i)_{i=1}^d$, and $\psi \in C_c^\infty(\mathbb{R}^d \times [0,1])$ is smooth and compactly supported.

We refer to Stepanov and Trevisan [2017, Theorem 3.4] for a rigorous discussion of this result in general metric spaces. Such results are often referred to as *superposition principles*. In the Euclidean setting, if one assumes that all measures admit a density with respect to some common dominating measure, it suffices to check the continuity equation directly, in which μ_t is replaced by a density p_t [Ambrosio et al., 2005, Villani, 2009].

Throughout this work, we assume all paths of measures and vector fields are sufficiently regular such that the superposition principle applies, i.e. it suffices to check the continuity equation to conclude whether a given path of measures is generated by a given vector field. In Theorem 1, we use the weak form of the continuity equation in order to construct a marginal vector field from conditional vector fields, such that this marginal vector field is guaranteed to generate our desired interpolating path of measures.

4 Function Space Flow Matching

Building on the notions in Section 3, we now introduce our Functional Flow Matching model (FFM). The Flow Matching model is a recently proposed continuous-time normalizing flow method developed for finite-dimensional spaces [Lipman et al., 2023, Chen and Lipman, 2023]. Our FFM approach builds on this earlier line of work to develop an extension of these methods to infinite-dimensional spaces. Proofs for all of our theorems are available in Appendix A.1.

The main technical challenge of generalizing the existing techniques to infinite-dimensional spaces is that existing methods rely heavily on the notion of a probability density function, either with respect to the Lebesgue measure in the case of a Euclidean space or with respect to the canonical volume measure on a Riemannian manifold. In infinite-dimensional (Banach) spaces, there does not exist an analogue of the Lebesgue measure – that is, any nonzero translation invariant Borel measure must assign infinite measure to any open set [Eldredge, 2016].

As such, our FFM model is necessarily posed in measure-theoretic terms. Our derivations shed light on strict requirements needed to obtain a well-posed model. For instance, we require an absolute continuity assumption between the conditional and marginal measures defined in Section 4.1. In Euclidean spaces, such assumptions are easy to satisfy, but are non-trivial in infinite-dimensional spaces, even for the simple setting of Gaussian measures (see Section 4.3). Moreover, our derivations demonstrate that naively applying a white-noise Gaussian measure (as is done in the Euclidean setting) leads to an ill-posed model in function space.

4.1 Constructing a Path of Measures

Suppose we associate to every $f \in \mathcal{F}$ a path of measures $(\mu_t^f)_{t \in [0,1]}$ such that $\mu_0^f = \mu_0$ is some fixed reference measure and μ_1^f is concentrated around f . For instance, μ_1^f could be a Gaussian measure with mean f and a covariance having small operator norm. We then marginalize over all such measures, where we mix over the data distribution ν . That is, we define a new probability measure $\mu_t \in \mathcal{P}(\mathcal{F})$ for $t \in [0, 1]$ via

$$\mu_t(A) = \int \mu_t^f(A) d\nu(f) \quad \forall A \in \mathcal{B}(\mathcal{F}). \quad (4)$$

Due to our conditions on μ_t^f , we then have that $\mu_0 = \mu_0$ and $\mu_1 \approx \nu$ is approximately the data distribution. Suppose further that each conditional path of measures μ_t^f is generated by some known vector field v_t^f . In the following theorem, we claim that we may construct a vector field v_t which generates the *marginal* path of measures μ_t from the conditional vector fields v_t^f .

Theorem 1.

Assume that $\int_0^1 \int_{\mathcal{F} \times \mathcal{F}} \|v_t^f(g)\| d\mu_t^f(g) d\nu(f) dt < \infty$. If $\mu_t^f \ll \mu_t$ for ν -almost every f and almost every $t \in [0, 1]$, then the vector field

$$v_t(g) = \int_{\mathcal{F}} v_t^f(g) \frac{d\mu_t^f(g)}{d\mu_t} d\nu(f) \quad (5)$$

generates the marginal path of measures $(\mu_t)_{t \in [0,1]}$ specified by Equation (4). That is, $(v_t)_{t \in [0,1]}$ and $(\mu_t)_{t \in [0,1]}$ solve the continuity equation (2). Here, $d\mu_t^f / d\mu_t$ is the Radon-Nikodym derivative of the conditional measure with respect to the marginal.

If this vector field v_t were known, we could generate samples by solving the corresponding flow ODE (Equation (1)) with initial condition $f \sim \mu_0$ drawn from our fixed reference measure. However, the vector field specified by Equation (5) is intractable. Thus, we will learn a model to approximate this unknown vector field. Note that our model will necessarily be a mapping between infinite dimensional spaces. We discuss how to parametrize such a model in Section 5.

The main technical assumption in Theorem 1 is that the conditional distributions μ_t^f are ν -almost surely absolutely continuous with respect to the marginal distribution μ_t . Although this assumption is not generally true even in the Euclidean setting, we prove in Theorem (2) that this assumption holds under an additional equivalency condition on the conditional measures. In Section 4.3, we discuss how this equivalency assumption may be satisfied under a Gaussian parametrization.

Theorem 2.

Consider a probability measure ν on \mathcal{F} and a collection of measures μ_t^f parametrized by $f \in \mathcal{F}$. Suppose that the collection of parametrized measures are ν -a.e. mutually absolutely continuous. Define the marginal measure μ_t via Equation (4). Then, $\mu_t^f \ll \mu_t$ for ν -a.e. f .

4.2 Special Case: Gaussian Measures

In this section, we specialize to the setting where the reference measure μ_0 and conditional measures μ_t^f are chosen to be Gaussian measures [Bogachev, 1998]. We make this ansatz for several reasons. Foremost, our marginal vector field (Equation (5)) requires an absolute continuity assumption. In infinite-dimensional (separable) Banach spaces, the absolute continuity of Gaussian measures is well-understood, e.g. via the Cameron-Martin theorem and the Feldman-Hájek theorem [Da Prato and Zabczyk, 2014, Bogachev, 1998]. Moreover, we are able to parametrize our Gaussian measures via Gaussian processes [Rasmussen and Williams, 2006, Wild et al., 2022] for which a number of flexible choices of kernels have been explored in the machine learning literature.

More formally, for any $f \in \mathcal{F}$ we define a conditional path of probability measures $(\mu_t)_{t \in [0,1]}$ to be a Gaussian measure $\mu_t^f = \mathcal{N}(m_t^f, C_t^f)$ with mean $m_t^f \in \mathcal{F}$ and

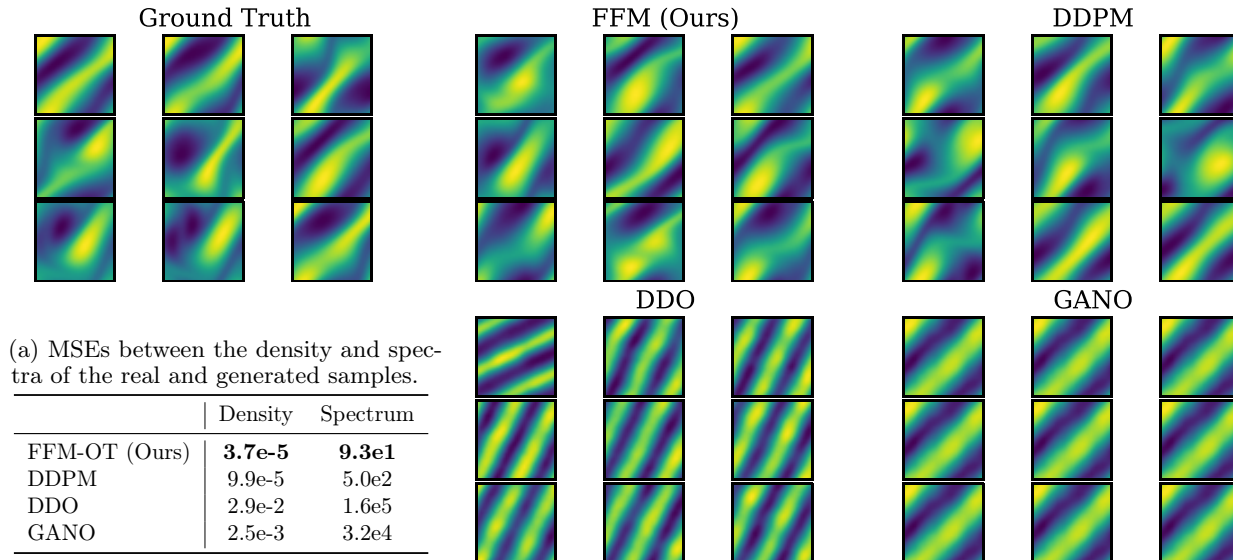


Figure 2: Samples from the Navier-Stokes dataset (“ground truth”) and samples from the various “models” considered in this work. Our FFM-OT model and DDPM qualitatively match the ground truth samples, whereas DDO and GANO suffer from mode collapse. Table 2a compares the density and spectra between 1000 real and generated samples, showing that our proposed method outperforms the others by a large margin on pointwise metrics. Note that we do not study the FFM-VP parametrization on this dataset due to computational costs.

covariance operator $C_t^f : \mathcal{F} \rightarrow \mathcal{F}$. Note that the C_t^f are necessarily symmetric, non-negative and trace-class [Da Prato and Zabczyk, 2014, Ch. 2]. In particular, this rules out multiples of the identity operator (corresponding to white noise) as a valid choice for C_t^f , as these operators are not compact and hence not trace-class.

In practice, we parametrize $t \mapsto m_t^f$ by a Fréchet differentiable mapping and specify C_t^f by a covariance operator C_0 and variance schedule $t \mapsto \sigma_t^f \in \mathbb{R}_{>0}$ such that $C_t^f = (\sigma_t^f)^2 C_0$. At time $t = 0$, we choose to parametrize $\mu_0^f = \mu_0 = \mathcal{N}(0, C_0)$ as a centered Gaussian measure independent of the function $f \in \mathcal{F}$. The measure μ_0 will serve as the reference measure in our generative model. In order to satisfy the desiderata of Section 4.1, at time $t = 1$ we will choose $m_1^f = f$ and C_1^f to have small operator norm so that μ_1^f is a Gaussian measure concentrated around f .

In this case, we note that the conditional flow $\phi^f : [0, 1] \times \mathcal{F} \rightarrow \mathcal{F}$ defined via $\phi_t^f(g) = \sigma_t^f g + m_t^f$ will push $g \sim \mathcal{N}(0, C_0)$ to the desired conditional measure μ_t^f , i.e. $\mu_t^f = [\phi_t^f]_{\#} \mathcal{N}(0, C_0)$. Using the flow ODE (Equation (1)), we see that a vector field generating this conditional path of measures is

$$v_t^f(g) = \frac{(\sigma_t^f)'}{\sigma_t^f} (g - m_t^f) + \frac{d}{dt} m_t^f \quad (6)$$

where $(\sigma_t^f)'$ is the ordinary time derivative of the variance schedule and $d/dt(m_t^f)$ is the Fréchet derivative

of the mapping $t \mapsto m_t^f$. The proof of this fact is a straightforward generalization of Lipman et al. [2023, Theorem 3], which demonstrates the analogous relationship in finite-dimensional Euclidean spaces.

In this work, we consider two concrete parameterizations. In the first parametrization (“OT”), the mean and variance are given as affine functions of t and f :

$$m_t^f = tf \quad \sigma_t^f = 1 - (1 - \sigma_{\min})t. \quad (7)$$

The “OT” path is named as such as it corresponds to an optimal transport map between Gaussians in the Euclidean setting [Lipman et al., 2023, McCann, 1997].

In the second parametrization (“VP”), we set

$$m_t^f = \alpha_{1-t} f \quad \sigma_t^f = \sqrt{1 - \alpha_{1-t}^2}. \quad (8)$$

This path is inspired by probability paths defined via variance preserving diffusion models [Lipman et al., 2023, Song et al., 2021]. We additionally experimented with the “variance exploding” parametrization [Lipman et al., 2023, Song et al., 2021], but found empirically that this was not suitable for our setting. See Appendix A.2 for details. Here, $\sigma_{\min} \in \mathbb{R}_{>0}$ and $\alpha_t \in \mathbb{R}_{>0}$ are hyperparameters of the model controlling the variance of the conditional measures.

4.3 Absolute Continuity for Gaussians

In general, the absolute continuity assumption of Theorem 1 is difficult to satisfy in function spaces. In the Gaussian setting, we may reduce this assumption to assumptions regarding the parametrization of our Gaussian measures. By the Feldman-Hájek theorem [Da Prato and Zabczyk, 2014, Theorem 2.25], our conditional Gaussian measures μ_t^f will be mutually absolutely continuous if the difference in means lies in the Cameron-Martin space of C_t , i.e. $m_t^f - m_t^g \in C_t^{1/2}(\mathcal{F})$.

Thus, under suitable assumptions on the data distribution ν and an appropriate parametrization of the conditional means, our marginal vector fields (Equation (5)) will be well-defined as a consequence of Theorem 2. Suppose $C_t = \sigma_t^2 C_0$ is a scaled version of some fixed covariance operator C_0 with the assumption that $0 < \sigma_t^2 \leq M$ is positive and bounded above. By Lemma 6.15 of Stuart [2010], this choice guarantees us that the Cameron-Martin space is constant in time, i.e. $C_0^{1/2}(\mathcal{F}) = C_t^{1/2}(\mathcal{F})$ for all $t \in [0, 1]$.

Assume further that the data distribution is supported on the Cameron-Martin space of C_0 , i.e. $\nu(C_0^{1/2}(\mathcal{F})) = 1$. In this case, given our covariance parametrization, our Gaussian measures will be mutually absolutely continuous if e.g. m_t^f is an affine function of f . We note that the parametrizations suggested in Section 4.2 are all affine, and so under the assumption that the data is supported on the Cameron-Martin space $C_0^{1/2}(\mathcal{F})$ our setup is well-defined.

In practice, verifying whether the data distribution is supported on $C_0^{1/2}(\mathcal{F})$ is difficult. One option to guarantee this assumption is satisfied is to pre-process the data via some mapping $T : \mathcal{F} \rightarrow C_0^{1/2}(\mathcal{F}) \subseteq \mathcal{F}$ whose image is contained in $C_0^{1/2}(\mathcal{F})$. We refer to Appendix C of Lim et al. [2023a] for a further discussion of such mappings and related results. We note that in practice, we do not find it necessary to perform this pre-processing.

4.4 Training the FFM Model

Ideally, we would like to perform functional regression on the marginal vector field defined via Equation (5), where we approximate $v_t(g)$ by a model $u_t(g | \theta)$ with parameters $\theta \in \mathbb{R}^p$. This could be achieved, for instance, by minimizing the loss

$$\mathcal{L}(\theta) = \mathbb{E}_{t \sim \mathcal{U}[0,1], g \sim \mu_t} \left[\|v_t(g) - u_t(g | \theta)\|^2 \right] \quad (9)$$

where $\mathcal{U}[0, 1]$ denotes a uniform distribution over the interval $[0, 1]$. Note here that our model is a mapping $u : \mathbb{R}^p \times [0, 1] \times \mathcal{F} \rightarrow \mathcal{F}$, i.e. our model is a parametrized, time-dependent operator on the func-

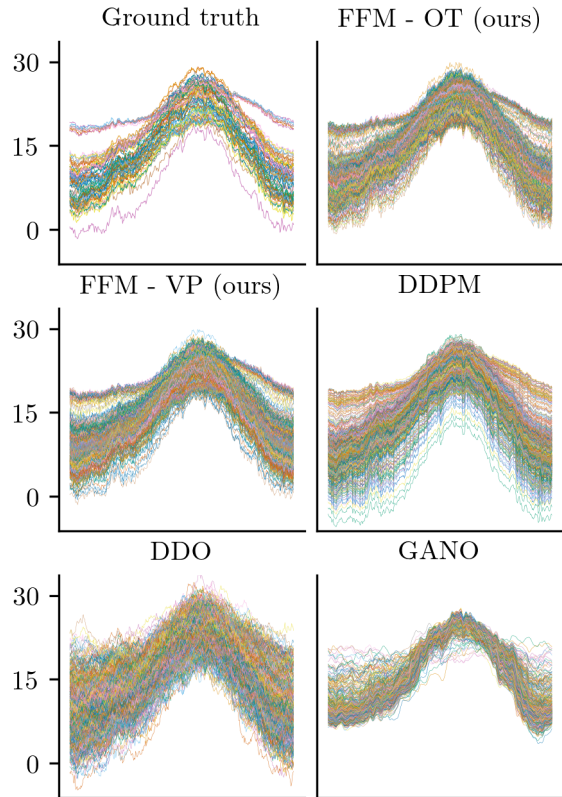


Figure 3: Unconditional generation of 500 samples on the AEMET dataset. Samples from our FFM model and DDPM appear visually to better match the characteristics of the real data relative to DDO and GANO.

tion space \mathcal{F} . However, such a loss is intractable to compute – in fact, if we had access to $(v_t)_{t \in [0,1]}$, there would be no need to learn a model. Consider instead the conditional loss, defined via

$$\mathcal{J}(\theta) = \mathbb{E}_{t \sim \mathcal{U}[0,1], f \sim \nu, g \sim \mu_t^f} \left[\left\| v_t^f(g) - u_t(f | \theta) \right\|^2 \right] \quad (10)$$

where, rather than regressing on the intractable v_t , we regress on the *known* conditional vector fields v_t^f . In the following theorem, we claim that minimizing $\mathcal{J}(\theta)$ is equivalent to minimizing $\mathcal{L}(\theta)$.

Theorem 3.

Assume that the true and model vector fields are square-integrable, i.e. $\int_0^1 \int_{\mathcal{F}} \|v_t^1(g)\|^2 d\mu_t(g) dt < \infty$ and $\int_0^1 \int_{\mathcal{F}} \|u_t(g | \theta)\|^2 d\mu_t(g) dt < \infty$. Then, $\mathcal{L}(\theta) = \mathcal{J}(\theta) + C$ where $C \in \mathbb{R}$ is a constant independent of θ .

5 Experiments

We now investigate the empirical performance of FFM on several real-world datasets. In all settings, we assume we are working in the space $\mathcal{F} = L^2([0, 1])$ and we parametrize $u_t(- | \theta)$ via a Fourier Neural Operator (FNO) [Li et al., 2021]. Sampling is achieved by drawing a sample from the reference measure $g \sim \mu_0$ and numerically solving the flow ODE (Equation (1)) with initial condition g . In our implementation, we use the DOPRI solver [Dormand and Prince, 1980]. Details can be found in Appendix A.2.¹

Datasets Our experiments in 1D include five datasets selected for their diverse correlation structures, exhibiting distinctive patterns that enable visual evaluation of generated samples. Plots of original and generated samples, as well as a detailed description of each dataset, can be found in Appendix A.2.2. The first dataset (AEMET) consists of a set of 73 curves describing the mean daily temperature at various locations [Febrero-Bande and de la Fuente, 2012]. The second is a gene expression time series dataset [Orlando et al., 2008], and the remaining three consist of global economic time series on population, GDP per capita, and labor force size [Bolt and Van Zanden, 2020, Inklaar et al., 2018, International Financial Statistics, 2022]. We also experiment with a dataset of solutions to the Navier-Stokes equation on a 2D torus [Li et al., 2022].

Baselines We compare against several functional generative models: the Denoising Diffusion Operator (DDO) [Lim et al., 2023a] with NCSN noise scale, GANO [Rahman et al., 2022], and functional DDPM [Kerrigan et al., 2023]. We do not compare to non-functional methods, as we are primarily interested in developing discretization-invariant generative models.

All noise was specified via a Gaussian process with a Matérn kernel where the lengthscale and variance is tuned for each dataset and method. Generally, tuning the parameters of the kernel is key to obtaining high-quality results across all models considered.

For the sake of a fair comparison, we used the same neural architecture for all models, with the exception of GANO which requires a generator and discriminator pair. We used the code provided by the authors of DDPM and GANO but re-implemented the DDO model. For all models, we performed extensive hyperparameter tuning and report the best results. Generally, we find the FFM methods are less sensitive to hyperparameter choices than the baseline methods.

¹Code for all of our experiments is available at github.com/GavinKerrigan/functional_flow_matching

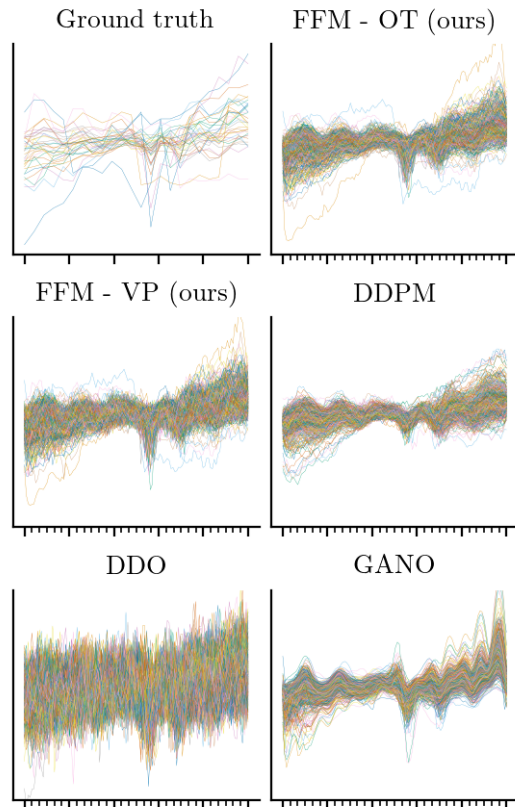


Figure 4: Samples from the Labor dataset and samples from the various models at 5x super-resolution.

Results Figure 2 shows samples from the Navier-Stokes dataset and samples generated from the various models we consider. Qualitatively, our FFM model and the DDPM model match the ground-truth samples, whereas DDO and GANO suffer from mode collapse.

Figure 3 shows samples from the AEMET dataset and generated samples from the models we consider. Our FFM model is able to qualitatively match the samples from the ground truth distribution. The DDPM samples are similar in quality, but do not respect the range of values seen in the data. For DDO, we observe smoothness issues, and for GANO, we again see mode collapse issues.

Quantitatively, Table 1 evaluates model performance on the one-dimensional datasets by computing pointwise statistics of the generated functions and computing the MSE between these pointwise statistics and those of the real data. Table 2a reports the MSE between the density and spectra [Lim et al., 2023b] of the real and generated samples on the Navier-Stokes dataset. See Appendix A.3 for visualizations.

Variants of FFM perform the best, on average, in almost all metrics considered across the wide range of domains

Functional Flow Matching

Table 1: Average MSEs between true and generated samples for pointwise statistics on five one-dimensional datasets, along with the standard deviation across ten random seeds. The average number of function evaluations (NFEs) for each sampling procedure in our implementation is also reported. Our FFM models obtain the best average performance across nearly all metrics, while simultaneously requiring fewer NFEs than the diffusion baselines.

		Mean	Variance	Skewness	Kurtosis	Autocorrelation	NFEs
AEMET	FFM-OT (ours)	8.4e-2 (9.9e-2)	1.7e+0 (1.1e+0)	7.7e-2 (6.6e-2)	3.3e-2 (3.7e-2)	3.0e-6 (4.0e-6)	668
	FFM-VP (ours)	1.3e-1 (1.4e-1)	1.5e+0 (1.2e+0)	5.e-2 (4.3e-2)	1.7e-2 (1.6e-2)	6.0e-6 (7.0e-6)	488
	DDPM	3.0e-1 (3.0e-1)	3.5e+0 (4.6e+0)	2.2e-1 (2.2e-1)	4.8e-2 (3.7e-2)	1.2e-5 (9.e-6)	1000
	DDO	2.4e-1 (1.4e-1)	6.6e+0 (5.1e+0)	2.1e-1 (4.1e-2)	3.8e-2 (1.3e-2)	6.7e-4 (1.3e-4)	2000
	GANO	6.5e+1 (1.9e+2)	3.7e+1 (4.0e+1)	2.9e+0 (4.8e+0)	3.3e-1 (4.0e-1)	1.2e-3 (3.1e-3)	1
Genes	FFM-OT (ours)	6.7e-4 (4.5e-4)	3.9e-3 (2.6e-4)	2.4e-1 (4.7e-2)	7.7e-2 (9.0e-3)	2.5e-4 (1.7e-4)	386
	FFM-VP (ours)	4.2e-4 (3.8e-4)	7.3e-4 (3.5e-4)	1.9e-1 (6.1e-2)	4.3e-2 (1.1e-2)	1.3e-4 (1.0e-4)	290
	DDPM	8.8e-4 (4.5e-4)	1.9e-3 (4.2e-4)	3.6e-1 (1.9e-1)	6.3e-2 (1.1e-2)	4.3e-4 (9.3e-5)	1000
	DDO	4.2e-3 (1.5e-3)	1.2e-3 (3.6e-4)	3.0e-1 (5.7e-2)	1.1e-1 (1.1e-2)	1.0e-3 (1.7e-4)	2000
	GANO	4.6e-3 (2.0e-3)	7.4e-3 (1.5e-3)	1.7e+0 (1.3e+0)	3.3e-1 (8.4e-2)	2.e-3 (1.0e-3)	1
Pop.	FFM-OT (ours)	3.9e-5 (3.8e-5)	7.0e-6 (9.e-6)	4.1e+0 (5.3e+0)	9.0e-2 (1.0e-1)	2.7e-5 (4.6e-5)	662
	FFM-VP (ours)	6.3e-5 (4.5e-5)	7.0e-6 (7.e-6)	1.3e+0 (6.1e-1)	7.8e-2 (4.5e-2)	2.5e-3 (5.2e-4)	494
	DDPM	5.7e-5 (5.2e-5)	6.0e-6 (7.0e-6)	1.9e+0 (1.2e+0)	5.9e-2 (4.4e-2)	5.6e-5 (3.5e-5)	1000
	DDO	1.9e-4 (8.7e-5)	2.7e-4 (1.9e-5)	4.2e+0 (4.1e-1)	2.7e-1 (3.7e-2)	3.2e-2 (1.9e-3)	2000
	GANO	1.1e-3 (9.8e-4)	4.3e-5 (7.1e-5)	8.e+0 (2.4e+0)	8.6e-1 (5.3e-1)	1.6e-3 (3.6e-3)	1
GDP	FFM-OT (ours)	2.0e-5 (1.2e-5)	9.e-6 (6.e-6)	6.3e-1 (3.5e-1)	3.9e-2 (1.9e-2)	2.8e-5 (1.4e-5)	536
	FFM-VP (ours)	4.1e-5 (2.1e-5)	8.0e-6 (7.0e-6)	6.2e-1 (4.1e-1)	5.0e-2 (2.5e-2)	1.9e-4 (2.3e-5)	494
	DDPM	1.6e-4 (1.5e-4)	2.5e-5 (2.9e-5)	8.6e-1 (5.9e-1)	5.1e-2 (2.1e-2)	1.4e-4 (1.0e-4)	1000
	DDO	2.1e-4 (1.1e-4)	2.9e-4 (9.4e-5)	1.7e+0 (1.1e-1)	2.7e-1 (2.4e-2)	9.6e-3 (1.5e-3)	2000
	GANO	8.4e-4 (7.8e-4)	5.0e-5 (3.7e-5)	2.6e+0 (1.3e+0)	2.1e-1 (1.4e-1)	1.6e-4 (1.6e-4)	1
Labor	FFM-OT (ours)	6.9e-5 (6.1e-5)	2.6e-5 (1.1e-5)	5.4e+0 (3.3e+0)	1.5e-1 (1.8e-1)	1.3e-4 (7.5e-5)	308
	FFM-VP (ours)	7.1e-5 (5.5e-5)	2.1e-5 (9.0e-6)	2.0e+0 (1.5e+0)	8.6e-2 (7.3e-2)	5.8e-4 (1.4e-4)	302
	DDPM	4.2e-4 (3.3e-4)	3.5e-4 (5.6e-4)	1.8e+3 (3.5e+3)	1.0e+1 (1.5e+1)	2.9e-4 (1.6e-4)	1000
	DDO	3.1e-4 (1.9e-4)	4.0e-4 (1.2e-4)	4.8e+0 (5.3e-1)	4.3e-1 (3.9e-2)	7.8e-3 (1.2e-3)	2000
	GANO	3.2e-3 (6.3e-3)	6.5e-4 (4.6e-4)	7.8e+0 (7.6e+0)	1.2e+0 (3.7e-1)	1.8e-3 (9.4e-4)	1

on which we performed evaluation. While pointwise statistics have limitations, for functional models there are no clear alternatives for evaluation, and pointwise metrics are broadly used in the literature [Rahman et al., 2022, Lim et al., 2023a]. Together with the qualitative results, these metrics further validate the performance of our method.

A key benefit of the FNO architecture is the ability to perform generation at arbitrary resolutions, a necessary component in any functional task. We demonstrate this on the Labor dataset in Figure 4. All models are trained on the original data resolution, but samples are drawn at a five times greater resolution. Samples from FFM and DDPM qualitatively match the characteristics of the ground truth distribution, whereas samples from DDO and GANO do not match the smoothness of the original data. See Appendix A.3 for further evaluation.

Conditional Generation We also demonstrate an extension of our method for conditional tasks, such as interpolating (or extrapolating) a finite set of given observations. We explore two approaches: conditional training and a modified sampling process inspired by ILVR [Choi et al., 2021]. We note alternative conditional methods [Mathieu et al., 2023] are readily applicable as well. In Figure 5, we demonstrate these two approaches. See Appendix A.4 for details.

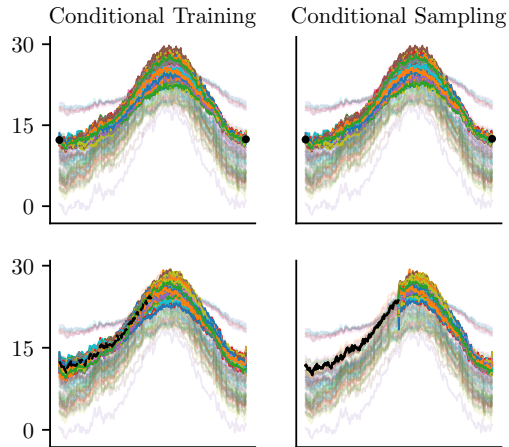


Figure 5: Conditional samples from the FFM-OT model. Darker curves indicate samples and lighter curves depict real data. Conditioning information is shown in black. The first column corresponds to a conditionally trained model and the second column corresponds to a conditionally sampled model in addition to conditional sampling. We see that, while the conditionally trained model takes into account the conditioning information, the conditional sampling method allows us to enforce equality of the generated samples to the conditioning information at the observation locations.

6 Conclusion

We introduce Functional Flow Matching (FFM), a continuous-time normalizing flow model which allows us to model infinite-dimensional distributions. We demonstrate that FFM is able to outperform several recently proposed function space generative models in terms of qualitative samples and pointwise metrics on a diverse set of benchmarks. Our work builds the foundations for function space normalizing flows, and our hope is that future work may build on these foundations. In terms of limitations, FFM is implemented via the FNO [Li et al., 2021], which can only handle data observed on uniform grids. Exploring architectures which alleviate this assumption may increase the applicability of our methods. Additionally, there are no established benchmarks for functional generation, unlike FID [Heusel et al., 2017] for images. Developing benchmarks for these tasks is critical for future work.

Acknowledgements. This research was supported in part by the Hasso Plattner Institute (HPI) Research Center in Machine Learning and Data Science at the University of California, Irvine, by NSF under awards 1900644 and 1927245, and by the National Institute of Health under awards R01-AG065330-02S1 and R01-LM013344.

References

- Michael Samuel Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *International Conference on Learning Representations*, 2023.
- Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient Flows: In Metric Spaces and in the Space of Probability Measures*. Springer Science & Business Media, 2005.
- Heli Ben-Hamu, Samuel Cohen, Joey Bose, Brandon Amos, Maximillian Nickel, Aditya Grover, Ricky T. Q. Chen, and Yaron Lipman. Matching normalizing flows and probability paths on manifolds. In *International Conference on Machine Learning*, volume 162, pages 1749–1763, 2022.
- Vladimir Igorevich Bogachev. *Gaussian Measures*. American Mathematical Society, 1998.
- Jutta Bolt and Jan Luiten Van Zanden. Maddison style estimates of the evolution of the world economy. A new 2020 update. *Maddison Project Working Paper*, 15, 2020.
- Ricky T. Q. Chen. torchdiffeq, 2018. URL <https://github.com/rtqichen/torchdiffeq>.
- Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31, 2018.
- Ricky TQ Chen and Yaron Lipman. Riemannian flow matching on general geometries. *arXiv preprint arXiv:2302.03660*, 2023.
- Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. ILVR: conditioning method for denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14367–14376, 2021.
- Giuseppe Da Prato and Jerzy Zabczyk. *Stochastic Equations in Infinite Dimensions*. Cambridge University Press, 2014.
- Aram Davtyan, Sepehr Sameni, and Paolo Favaro. Randomized conditional flow matching for video prediction. *arXiv preprint arXiv:2211.14575*, 2022.
- Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems*, volume 34, pages 8780–8794, 2021.
- J.R. Dormand and P.J. Prince. A family of embedded Runge-Kutta formulae. *Journal of Computational and Applied Mathematics*, 6:19–26, 1980.
- Nathaniel Eldredge. Analysis and probability on infinite-dimensional spaces. *arXiv preprint arXiv:1607.03591*, 2016.
- 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.
- Giulio Franzese, Simone Rossi, Dario Rossi, Markus Heinonen, Maurizio Filippone, and Pietro Michiardi. Continuous-time functional diffusion processes. *arXiv preprint arXiv:2303.00800*, 2023.
- Karan Goel, Albert Gu, Chris Donahue, and Christopher Ré. It’s raw! Audio generation with state-space models. In *International Conference on Machine Learning*, pages 7616–7633, 2022.
- Paul Hagemann, Lars Ruthotto, Gabriele Steidl, and Nicole Tianjiao Yang. Multilevel diffusion: Infinite dimensional score-based diffusion models for image generation. *arXiv preprint arXiv:2303.04772*, 2023.
- Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (GELUs). *arXiv preprint arXiv:1606.08415*, 2016.
- Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

- Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851, 2020.
- Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, and Tim Salimans. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022.
- Robert Inklaar, Herman de Jong, Jutta Bolt, and Jan Luiten Van Zanden. Rebasings ’Maddison’: new income comparisons and the shape of long-run economic development. *GGDC Research Memorandum*, 174, 2018.
- International Financial Statistics. Prices, Production, and Labor, Labor Force, 2022. URL <https://data.imf.org/regular.aspx?key=63087884>.
- Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up GANs for text-to-image synthesis. *arXiv preprint arXiv:2303.05511*, 2023.
- Gavin Kerrigan, Justin Ley, and Padhraic Smyth. Diffusion generative models in infinite dimensions. In *International Conference on Artificial Intelligence and Statistics*, volume 206, pages 9538–9563, 2023.
- Patrick Kidger, James Foster, Xuechen Li, and Terry J Lyons. Neural SDEs as infinite-dimensional GANs. In *International Conference on Machine Learning*, volume 139, pages 5453–5463, 2021.
- Ivan Kobyzev, Simon J.D. Prince, and Marcus A. Brubaker. Normalizing flows: An introduction and review of current methods. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 43:3964–3979, 2020.
- Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. In *International Conference on Learning Representations*, 2021.
- Nikola B. Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew M. Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces. *CoRR*, abs/2108.08481, 2021.
- 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.
- Zongyi Li, Miguel Liu-Schiaffini, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Learning chaotic dynamics in dissipative systems. In *Advances in Neural Information Processing Systems*, 2022.
- Jae Hyun Lim, Nikola B Kovachki, Ricardo Baptista, Christopher Beckham, Kamyar Azizzadenesheli, Jean Kossaifi, Vikram Voleti, Jiaming Song, Karsten Kreis, Jan Kautz, Chrisotopher Pal, Arash Vahdat, and Anima Anandkumar. Score-based diffusion models in function space. *arXiv preprint arXiv:2302.07400*, 2023a.
- Jen Ning Lim, Sebastian Vollmer, Lorenz Wolf, and Andrew Duncan. Energy-based models for functional data using path measure tilting. In *International Conference on Artificial Intelligence and Statistics*, pages 1904–1923, 2023b.
- Lequan Lin, Zhengkun Li, Ruikun Li, Xuliang Li, and Junbin Gao. Diffusion models for time series applications: A survey. *arXiv preprint arXiv:2305.00624*, 2023.
- Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.
- Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.
- Emile Mathieu, Vincent Dutordoir, Michael J Hutchinson, Valentin De Bortoli, Yee Whye Teh, and Richard E Turner. Geometric neural diffusion processes. *arXiv preprint arXiv:2307.05431*, 2023.
- Robert J McCann. A convexity principle for interacting gases. *Advances in Mathematics*, 128(1):153–179, 1997.
- Kirill Neklyudov, Daniel Severo, and Alireza Makhzani. Action matching: A variational method for learning stochastic dynamics from samples. *arXiv preprint arXiv:2210.06662*, 2022.
- Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171, 2021.
- Donal O’Regan. *Differential Equations in Abstract Spaces*, pages 186–194. Springer Netherlands, 1997.
- David A Orlando, Charles Y Lin, Allister Bernard, Jean Y Wang, Joshua ES Socolar, Edwin S Iversen, Alexander J Hartemink, and Steven B Haase. Global control of cell-cycle transcription by coupled cdk and network oscillators. *Nature*, 453:944–947, 2008.
- George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic model-

- ing and inference. *The Journal of Machine Learning Research*, 22:2617–2680, 2021.
- 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. ISSN 2835-8856.
- Carl Edward Rasmussen and Christopher Williams. *Gaussian Processes for Machine Learning*. Springer, 2006.
- Kashif Rasul, Calvin Seward, Ingmar Schuster, and Roland Vollgraf. Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In *International Conference on Machine Learning*, pages 8857–8868, 2021a.
- Kashif Rasul, Abdul-Saboor Sheikh, Ingmar Schuster, Urs M Bergmann, and Roland Vollgraf. Multivariate probabilistic time series forecasting via conditioned normalizing flows. In *International Conference on Learning Representations*, 2021b.
- Yulia Rubanova, Ricky T. Q. Chen, and David K. Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. In *Advances in Neural Information Processing Systems*, volume 32, 2019.
- Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, 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*, volume 35, pages 36479–36494, 2022.
- Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, volume 32, 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.
- Eugene Stepanov and Dario Trevisan. Three superposition principles: currents, continuity equations and curves of measures. *Journal of Functional Analysis*, 272(3):1044–1103, 2017.
- Andrew M. Stuart. Inverse problems: a Bayesian perspective. *Acta Numerica*, 19:451–559, 2010.
- 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*, volume 34, pages 24804–24816, 2021.
- Cédric Villani. *Optimal Transport: Old and New*. Springer Berlin Heidelberg, 2009.
- Veit David 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*, volume 35, pages 3716–3730, 2022.
- Tijin Yan, Hongwei Zhang, Tong Zhou, Yufeng Zhan, and Yuanqing Xia. Scoregrad: Multivariate probabilistic time series forecasting with continuous energy-based generative models. *arXiv preprint arXiv:2106.10121*, 2021.
- Cagatay Yildiz, Markus Heinonen, and Harri Lahdesmaki. ODE2VAE: Deep generative second order ODEs with Bayesian neural networks. In *Advances in Neural Information Processing Systems*, volume 32, 2019.
- Jinsung Yoon, Daniel Jarrett, and Mihaela van der Schaar. Time-series generative adversarial networks. In *Advances in Neural Information Processing Systems*, volume 32, 2019.
- Samuel Zaidman. *Functional Analysis and Differential Equations in Abstract Spaces*. Taylor & Francis, 1999.
- Linqi Zhou, Michael Poli, Winnie Xu, Stefano Marsaroli, and Stefano Ermon. Deep latent state space models for time-series generation. *arXiv preprint arXiv:2212.12749*, 2022.

Checklist

1. For all models and algorithms presented, check if you include:
 - (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. **[Yes]**
 - (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. **[Not Applicable]**
 - (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. **[Yes]**
2. For any theoretical claim, check if you include:
 - (a) Statements of the full set of assumptions of all theoretical results. **[Yes]**
 - (b) Complete proofs of all theoretical results. **[Yes]**
 - (c) Clear explanations of any assumptions. **[Yes]**
3. For all figures and tables that present empirical results, check if you include:
 - (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). **[Yes]**
 - (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). **[Yes]**
 - (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). **[Yes]**
 - (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). **[Yes]**
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
 - (a) Citations of the creator If your work uses existing assets. **[Yes]**
 - (b) The license information of the assets, if applicable. **[Yes]**
 - (c) New assets either in the supplemental material or as a URL, if applicable. **[Not Applicable]**
 - (d) Information about consent from data providers/curators. **[Not Applicable]**
 - (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. **[Not Applicable]**
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
 - (a) The full text of instructions given to participants and screenshots. **[Not Applicable]**
 - (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. **[Not Applicable]**
 - (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. **[Not Applicable]**

Functional Flow Matching: Appendix

A.1 Proofs of Theorems

Theorem 1.

Assume that $\int_0^1 \int_{\mathcal{F} \times \mathcal{F}} \|v_t^f(g)\| d\mu_t^f(g) d\nu(f) dt < \infty$. If $\mu_t^f \ll \mu_t$ for ν -almost every f and almost every $t \in [0, 1]$, then the vector field

$$v_t(g) = \int_{\mathcal{F}} v_t^f(g) \frac{d\mu_t^f}{d\mu_t}(g) d\nu(f) \quad (5)$$

generates the marginal path of measures $(\mu_t)_{t \in [0,1]}$ specified by Equation (4). That is, $(v_t)_{t \in [0,1]}$ and $(\mu_t)_{t \in [0,1]}$ solve the continuity equation (2). Here, $d\mu_t^f/d\mu_t$ is the Radon-Nikodym derivative of the conditional measure with respect to the marginal.

Proof. In this proof, we denote the variable of integration for integrals over \mathcal{F} via a subscript on the integral. We show that for an arbitrary but fixed test function φ ,

$$\int_0^1 \int_g \partial_t \varphi(g, t) d\mu_t(g) dt = - \int_0^1 \int_g \langle v_t(g), \nabla_g \varphi(g, t) \rangle d\mu_t(g) dt. \quad (11)$$

To that end, we begin by analyzing the left-hand side, replacing the integration of the marginal measure μ_t with a double integral over its components:

$$\int_0^1 \int_g \partial_t \varphi(g, t) d\mu_t(g) dt = \int_0^1 \int_f \int_g \partial_t \varphi(g, t) d\mu_t^f(g) d\nu(f) dt \quad (12)$$

By Fubini-Tonelli and using the assumption that v_t^f generates μ_t^f , we obtain via the continuity equation for (v_t^f, μ_t^f) :

$$= - \int_f \int_0^1 \int_g \langle v_t^f(g), \nabla_g \varphi(g, t) \rangle d\mu_t^f(g) dt d\nu(f) \quad (13)$$

Using our absolute continuity assumption and Fubini-Tonelli once again, we perform a change of measure to obtain

$$= - \int_0^1 \int_f \int_g \langle v_t^f(g), \nabla_g \varphi(g, t) \rangle \left(\frac{d\mu_t^f}{d\mu_t}(g) \right) d\mu_t(g) d\nu(f) dt \quad (14)$$

$$= - \int_0^1 \int_f \int_g \langle v_t^f(g) \frac{d\mu_t^f}{d\mu_t}(g), \nabla_g \varphi(g, t) \rangle d\mu_t(g) d\nu(f) dt \quad (15)$$

Using the fact that Bochner integrals commute with inner products, an application of Fubini-Tonelli yields

$$= - \int_0^1 \int_{\mathcal{G}} \left\langle \int_{\mathcal{F}} v_t^f(g) \frac{d\mu_t^f}{d\mu_t}(g) d\nu(f), \nabla_f \varphi(f, t) \right\rangle d\mu_t(g) dt \quad (16)$$

$$= - \int_0^1 \int_{\mathcal{G}} \langle v_t(g), \nabla_f \varphi(f, t) \rangle d\mu_t(g) dt \quad (17)$$

Hence, we have shown that the vector field generating μ_t is given by

$$v_t(g) = \int_{\mathcal{F}} v_t^f(g) \frac{d\mu_t^f}{d\mu_t}(g) d\nu(f). \quad (18)$$

□

Theorem 2.

Consider a probability measure ν on \mathcal{F} and a collection of measures μ_t^f parametrized by $f \in \mathcal{F}$. Suppose that the collection of parametrized measures are ν -a.e. mutually absolutely continuous. Define the marginal measure μ_t via Equation (4). Then, $\mu_t^f \ll \mu_t$ for ν -a.e. f .

Proof. By assumption, there exists $\mathcal{G} \subseteq \mathcal{F}$ with $\nu(\mathcal{G}) = 1$ and for any $f, g \in \mathcal{G}$, we have $\mu_t^f \ll \mu_t^g$ and $\mu_t^g \ll \mu_t^f$. Fix $A \in \mathcal{B}(\mathcal{F})$ with $\mu_t(A) = 0$. We claim that $\mu^f(A) = 0$ for every $f \in \mathcal{G}$. Note that as $\mathcal{G} \subseteq \mathcal{F}$ has full measure, the integral defining μ_T may be taken over \mathcal{G} rather than \mathcal{F} . Suppose for the sake of contradiction that $\mu_t^f(A) > 0$ for some $f \in \mathcal{G}$. From the mutual equivalencies of the measures parametrized by \mathcal{G} , it follows that $\mu_t^g(A) > 0$ for every $g \in \mathcal{G}$. Given the form of the mixture measure μ_t , it would then follow that $\mu_t(A) > 0$, which is a contradiction. Thus, $\mu_t^f \ll \mu_t$ for ν -a.e. f as claimed. □

Theorem 3.

Assume that the true and model vector fields are square-integrable, i.e. $\int_0^1 \int_{\mathcal{F}} \|v_t(g)\|^2 d\mu_t(g) dt < \infty$ and $\int_0^1 \int_{\mathcal{F}} \|u_t(g | \theta)\|^2 d\mu_t(g) dt < \infty$. Then, $\mathcal{L}(\theta) = \mathcal{J}(\theta) + C$ where $C \in \mathbb{R}$ is a constant independent of θ .

Proof. First, note that since we are working in a real Hilbert space, for fixed $f, g \in \mathcal{F}$ we have

$$\|v_t(g) - u_t(g | \theta)\|^2 = \langle v_t(g) - u_t(g | \theta), v_t(g) - u_t(g | \theta) \rangle \quad (19)$$

$$= \|v_t(g)\|^2 + \|u_t(g | \theta)\|^2 - 2\langle v_t(g), u_t(g | \theta) \rangle \quad (20)$$

and similarly,

$$\|v_t^f(g) - u_t(g | \theta)\|^2 = \|v_t^f(g)\|^2 + \|u_t(g | \theta)\|^2 - 2\langle v_t^f(g), u_t(g | \theta) \rangle. \quad (21)$$

The first term in both is independent of the model parameters θ . We analyze the remaining two terms. Below, we use subscripts on integrals over \mathcal{F} to denote the variable of integration.

First, using the fact that μ_t is a mixture measure,

$$\mathbb{E}_{t, \mu_t} \left[\|u_t(g | \theta)\|^2 \right] = \quad (22)$$

$$= \int_0^1 \int_{\mathcal{G}} \|u_t(g | \theta)\|^2 d\mu_t(g) dt \quad (23)$$

$$= \int_0^1 \int_{\mathcal{F}} \int_{\mathcal{G}} \|u_t(g | \theta)\|^2 d\mu_t^f(g) d\nu(f) dt \quad (24)$$

$$= \mathbb{E}_{t, g \sim \mu_t^f, f \sim \nu} \left[\|u_t(g | \theta)\|^2 \right]. \quad (25)$$

Next, using the exchangeability between Bochner integrals and inner products and Fubini-Tonelli,

$$\mathbb{E}_{t,g \sim \mu_t} [\langle v_t(g), u_t(g | \theta) \rangle] = \tag{26}$$

$$= \int_0^1 \int_g \langle v_t(g), u_t(g | \theta) \rangle d\mu_t(g) dt \tag{27}$$

$$= \int_0^1 \int_g \left\langle \int_f v_t^f(g) \frac{d\mu_t^f}{d\mu_t}(g) d\nu(f), u_t(g | \theta) \right\rangle d\mu_t(g) dt \tag{28}$$

$$= \int_0^1 \int_f \int_g \langle v_t^f(g), u_t(g | \theta) \rangle \left(\frac{d\mu_t^f}{d\mu_t}(g) \right) d\mu_t(g) d\nu(f) dt \tag{29}$$

$$= \int_0^1 \int_f \int_g \langle v_t^f(g), u_t(g | \theta) \rangle d\mu_t^f(g) d\nu(f) dt \tag{30}$$

$$= \mathbb{E}_{t,f \sim \nu, g \sim \mu_t^f} [\langle v_t^f(g), u_t(g | \theta) \rangle]. \tag{31}$$

This shows the equivalency of the two losses. \square

A.2 Experiment Details

A.2.1 Parametrizations

The FM-OT and FM-VP model require specifying a variance schedule via the hyperparameter σ_t^f . In this work, we parametrize FFM-OT by setting $\sigma_{\min} = 1e-4$. For FFM-VP, we set $\alpha_t = \cos\left(\frac{t+s}{1+s}\frac{\pi}{2}\right)$ where $s = 0.08$, following a formulation similar to the cosine schedule introduced by Nichol and Dhariwal [2021].

Model-specific hyperparameters have been extensively fine-tuned via grid search, and we found the following parametrizations to consistently perform optimally across several domains:

- DDPM: the noise schedule, following the notation of Kerrigan et al. [2023], is set to linearly interpolate between $\beta_0 = 1e - 4$ and $\beta_T = 0.02$ in $T = 1000$ timesteps. The code for this implementation was taken directly from the official repository².
- DDO: following the notation of Lim et al. [2023a], we set the time interval to $T = 10$, and the noise schedule geometrically interpolates between $\sigma_{10} = 1e - 3$ and $\sigma_1 = 1$ on the 1D datasets and $\sigma_{10} = 1e - 2$ and $\sigma_1 = 100$ on the 2D datasets. Sampling is performed by running their annealed Langevin dynamics algorithm with $\epsilon = 2 \times 10^{-5}$ and $M = 200$.
- GANO: the generator is trained every 5 epochs, and gradient penalty set to $\lambda = 0.1$ in 1D and $\lambda = 10$ in 2D. [Rahman et al., 2022]. The code for this implementation was taken directly from the official repository³.

Model Architectures For FFM, DDPM, and DDO, the architecture used is the FNO implemented in the `neuraloperator` package [Li et al., 2021, Kovachki et al., 2021]. For GANO, we directly use the FNO-based model architectures for both the discriminator and generator implemented by Rahman et al. [2022] for the 2D dataset, while for the 1D datasets we use the same FNO architecture as the other methods.

Gaussian Measures Each model experimented with relies on noise sampled from a Gaussian measure. In our work, we consider a mean-zero Gaussian process (GP) parametrized by a Matérn kernel with $\nu = 1/2$. In 1D, the kernel hyperparameters are set to have a variance $\sigma^2 = 0.1$ and length scale $\ell = 1e - 2$. In 2D, the variance is $\sigma^2 = 1$ and the length scale was set to $\ell = 1e - 3$ for DDO and GANO, and $\ell = 1e - 2$ for FFM and DDPM.

²https://github.com/GavinKerrigan/functional_diffusion

³<https://github.com/neuraloperator/GANO>

Training All models are trained using the Adam optimizer. In 1D, we use an initial learning rate of $1e-3$, scheduled to decrease by one order of magnitude after 50 epochs for all datasets but AEMET, where it is decreased every 25 epochs. In 2D, we use an initial learning rate of $5e-4$ for FFM, DDPM, and DDO, and an initial learning rate of $1e-4$ for GANO, and this initial learning rate was decayed by one order of magnitude every 25 epochs.

A.2.2 Dataset Details

Navier-Stokes. This dataset consists of solutions to the Navier-Stokes equations on a 2D torus at a resolution of 64×64 . For the sake of efficient training, we randomly selected 20,000 datapoints for training from the original dataset [Li et al., 2022] as there is a high degree of redundancy in the data. For FFM, DDPM, and DNO, we use 4 Fourier layers of 32 modes and 64 hidden channels, 256 lifting channels, 256 projection channels, and the GeLU activation function [Hendrycks and Gimpel, 2016]. For GANO, we use 32 modes and set the number of hidden channels to 16 due to memory constraints. All models were trained for 300 epochs at a batch size of 128.

AEMET dataset. This dataset consists of a set of functions describing the mean curve of the average daily temperature (in Celsius) for the period 1980-2009 recorded by 73 weather stations in Spain [Febrero-Bande and de la Fuente, 2012]. Each function is observed on a uniform grid at a resolution of 365. The neural architecture we use for this dataset is an FNO with a width of 256 and 64 modes, kept constant for all models considered in this experiment. The model is trained for 50 epochs, with batch size set to 73.

Gene expression. The original dataset consists of 10,928 time series at 20 uniformly spaced time points, recording the amplitude of gene expression for 4 different genes. The genes are concatenated to create the visual effect of spikes occurring periodically in time, while maintaining the structure of the original dataset. The data was log-transformed and centered before being fed to the model. We restrict our focus to a subset of 156 functions exhibiting large gene expression, determined by the standard deviation averaged across time for each centered function being greater than 0.3. For this dataset, we use an FNO with a width of 256 and 16 modes across all models. The model is trained for 200 epochs, with batch size set to 16.

Economic datasets. The first two datasets are taken from the Maddison Project database [Bolt and Van Zanden, 2020, Inklaar et al., 2018], and the third from the IMF [International Financial Statistics, 2022]. The datasets were picked specifically for their distinct visual characteristics, explored in greater detail in Appendix A.3.

- *Population*: time series of the evolution of the population for 169 countries across the globe from the year 1950 to 2018 (that is, discretized at 69 points in time). For a clearer visual representation, each time series was divided by its mean, so each curve represents the population for each country, relative to the mean population for that country over the 69 years under consideration. The functions in this dataset exhibit linear growth over time, with a change point shared across observations.
- *GDP*: time series representing the evolution of GDP per capita from 1950 to 2018. The original dataset consists of 169 countries, but time series presenting missing values were removed yielding 145 observations. The same preprocessing as that described above was applied to the data. While the functions seem to exhibit the same change point as that observed for the population datasets, the growth over time is noisier and exhibits irregular patterns.
- *Labor*: size of labor force per quarter between Q1 2017 and Q4 2022 (for a total of 24 points in time), for a subset of 35 countries (obtained removing those with missing values from the original 105 observations). The same preprocessing as that described for the population dataset was applied to the data. This dataset tests the ability of the generative models to learn from small and multimodal data.

The models for the population and GDP datasets have width set to 256 and 32 modes, while the one for the labor dataset has width set to 128 and 8 modes. All models were trained for 100 epochs, with a batch size of 16.

A.2.3 Sampling Details

We use the `torchdiffeq` [Chen, 2018] package for all ODE solvers. The specific solver we use is `dopri5`, an implementation of the Dormand-Prince method [Dormand and Prince, 1980] of order 5. We set the absolute and

relative tolerance parameters to $1e-10$ for the 1D datasets and $1e-5$ for the 2D datasets. Note that setting such a tolerance gives us an explicit way of trading off sample quality for sampling efficiency.

A.3 Additional Experimental Results

This section contains several additional experimental results and figures. First, in Section A.3.1 we explore an additional synthetic dataset consisting of a mixture of Gaussian processes (MoGP). Next, we provide additional results on super-resolution in Section A.3.2. Sections A.3.3 and A.3.4 provide further visualizations of our results on the 1D and 2D datasets considered in the main paper respectively.

A.3.1 Additional Results: MoGP

Mixture of Gaussian Processes (MoGP) Dataset. The experiment considers the task of generating samples from a mixture of two GPs. The two components, with equal weights, have mean functions $m_1 = 10x - 5$ and $m_2 = -10x + 5$, and a squared-exponential kernel with variance $\sigma^2 = 0.04$ and length scale $\ell = 0.1$. The synthetic samples used for training are observed on a uniform grid at a resolution of 64 on the interval $[0, 1]$. All models were trained on the same sample of $N = 5000$ realizations of the mixture of GPs. Figure 6 illustrates a visual comparison of 500 samples from each model, while Table 2 presents a quantitative comparison of the mean squared error (MSE) on various pointwise statistics. Additionally, Figure 6 provides a comprehensive depiction of the variations in these pointwise statistics across different models.

Table 2: Average MSEs between true and generated samples for various pointwise statistics on the MoGP dataset, along with the standard deviation (across ten random seeds). The average number of function evaluations (NFEs) for each model is also reported. Variants of our proposed FFM model obtain the best or second best average performance across all metrics. DDPM outperforms FFM in terms of variance, but only by a small margin.

	Mean	Variance	Skewness	Kurtosis	Autocorrelation	NFEs
FFM-OT (ours)	2.2e-2 (3.e-2)	2.9e-1 (3.2e-1)	1.6e-2 (1.1e-2)	1.1e-2 (1.2e-2)	7.e-6 (6.e-6)	740
FFM-VP (ours)	3.9e-2 (3.6e-2)	3.6e-1 (5.6e-1)	1.4e-2 (5.2e-3)	1.5e-2 (1.2e-2)	8.e-6 (8.e-6)	716
DDPM	3.0e-2 (2.4e-2)	1.4e-1 (1.9e-1)	1.5e-2 (9.6e-3)	1.2e-2 (8.1e-3)	1.9e-5 (2.2e-5)	1000
DDO	7.3e-1 (9.6e-1)	2.7e+0 (5.3e+0)	4.2e-1 (8.7e-1)	2.8e-1 (3.9e-1)	1.3e-5 (8.e-6)	2000
GANO	1.9e-1 (1.6e-1)	8.1e+0 (6.0e+0)	3.4e-1 (2.5e-1)	4.6e-2 (3.9e-2)	6.2e-4 (6.7e-4)	1

A.3.2 Additional Results: Super-Resolution

Here, we provide additional visualizations regarding the ability of all models considered to perform super-resolution, i.e. to sample at a resolution greater than the training dataset. All models considered are trained at the original dataset resolution, but due to the neural operator architectures being used, we may sample at arbitrary resolutions. We note that quantitatively evaluating these super-resolved samples is difficult as we do not have access to a notion of higher-resolution ground truth here.

Figure 7 shows samples from the original Gene expression time series dataset [Orlando et al., 2008] at a resolution of 20, as well as samples from each model at a 5x resolution, i.e. a resolution of 100. We see that, qualitatively, samples from FFM, DDPM, and GANO resemble those of the original dataset. The samples from DDO appear overly rough, and the samples from GANO are smoother than those from FFM and DDPM.

Figure 8 shows qualitatively similar results on the econometrics datasets [Bolt and Van Zanden, 2020, Inklaar et al., 2018, International Financial Statistics, 2022], with the exception of FFM-VP generating samples that are rougher than the original data. To further explore the quality of these super-resolved samples, we additionally provide the correlation matrices of the original data and super-resolved samples in Figure 9. We generally see that FFM-OT, FFM-VP, DDPM, and GANO are able to qualitatively capture the original correlation structures, whereas DDO fails to do so. We additionally note that on the Population and GPD datasets, FFM-VP, DDO, and GANO display a consistent strong diagonal band, indicating that these models generate samples at a variance which is too large when super-resolved. All models display this failure mode on the Labor dataset.

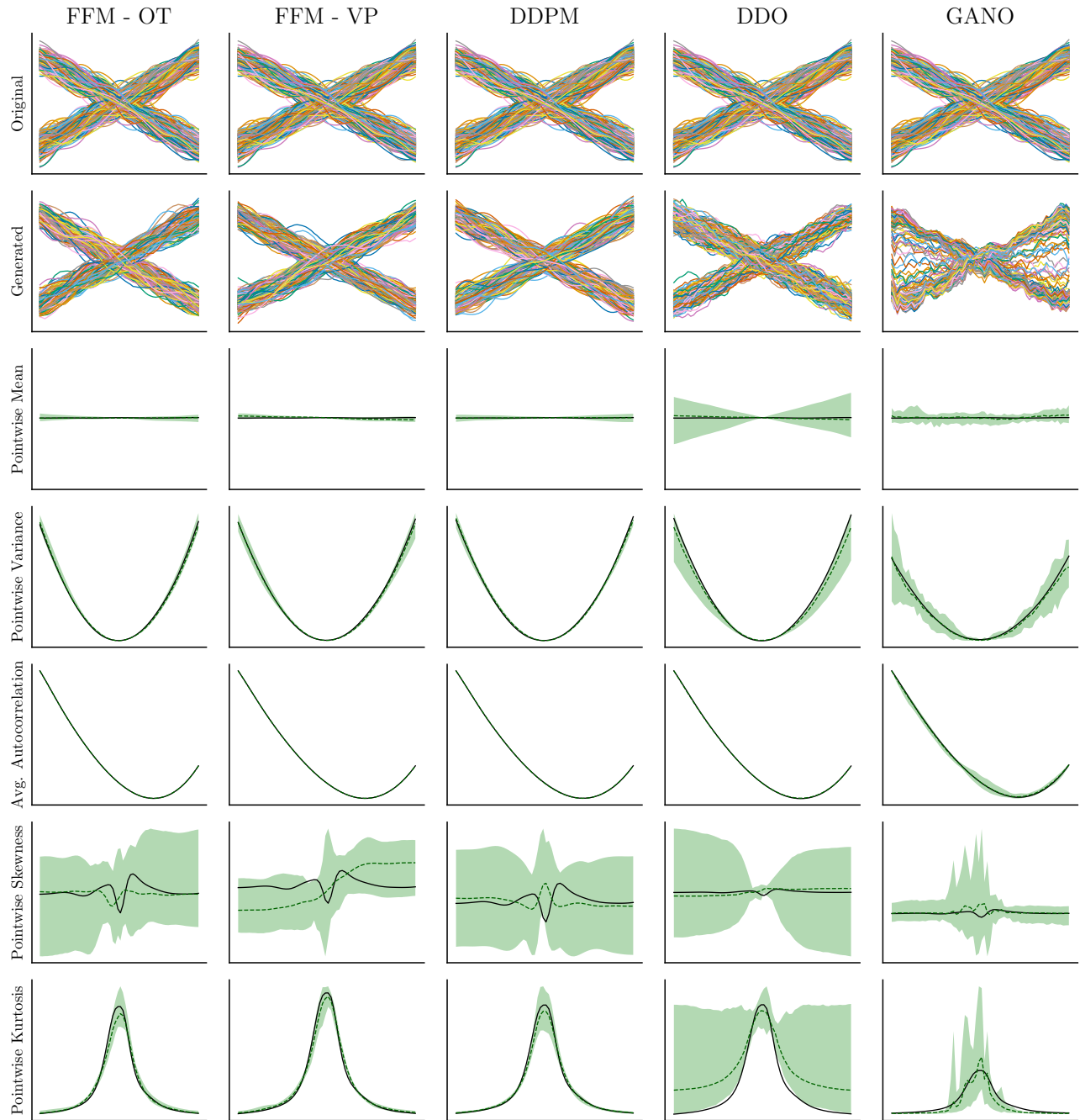


Figure 6: Various pointwise statistics on the MoGP dataset. Curves in black indicate the corresponding pointwise statistic for the original dataset (top row). The green error bands represent the minimal and maximal value of the pointwise statistic from 500 samples across ten random seeds of the corresponding model. The dashed green lines indicate the mean pointwise statistic across these ten runs for each model. See Table 2 for a quantitative comparison.

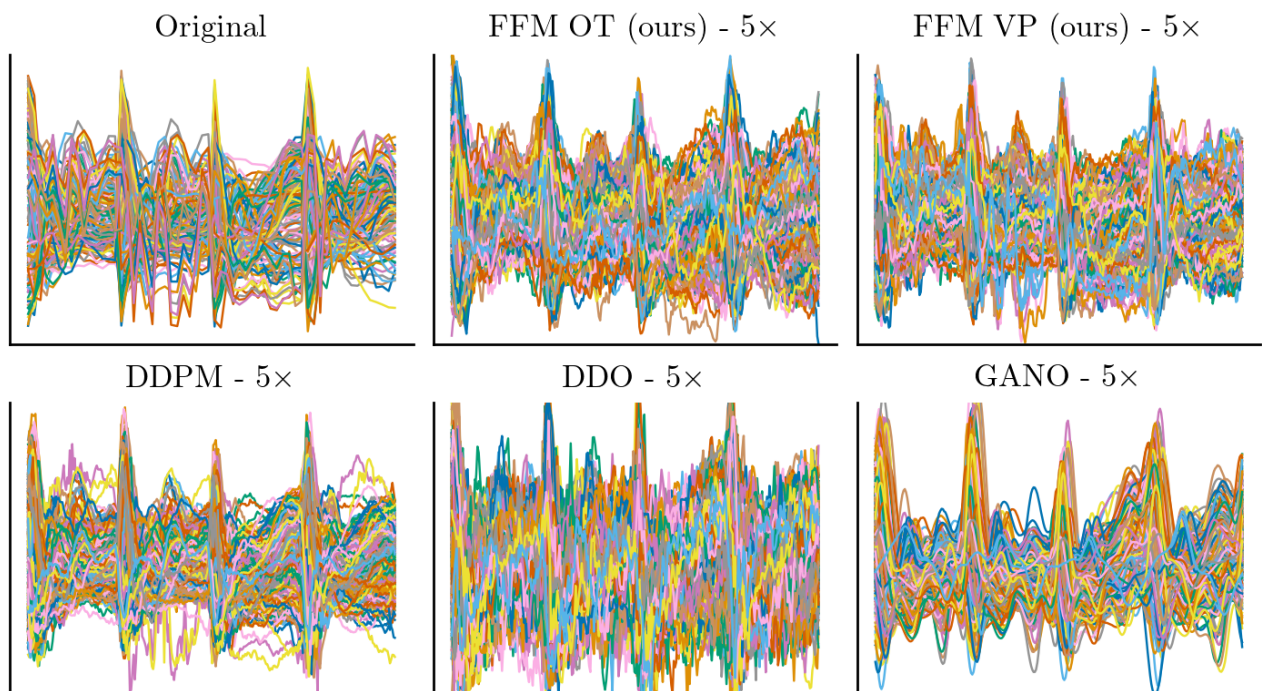


Figure 7: Samples from the gene expression dataset and samples from the various models at a 5x super-resolution.

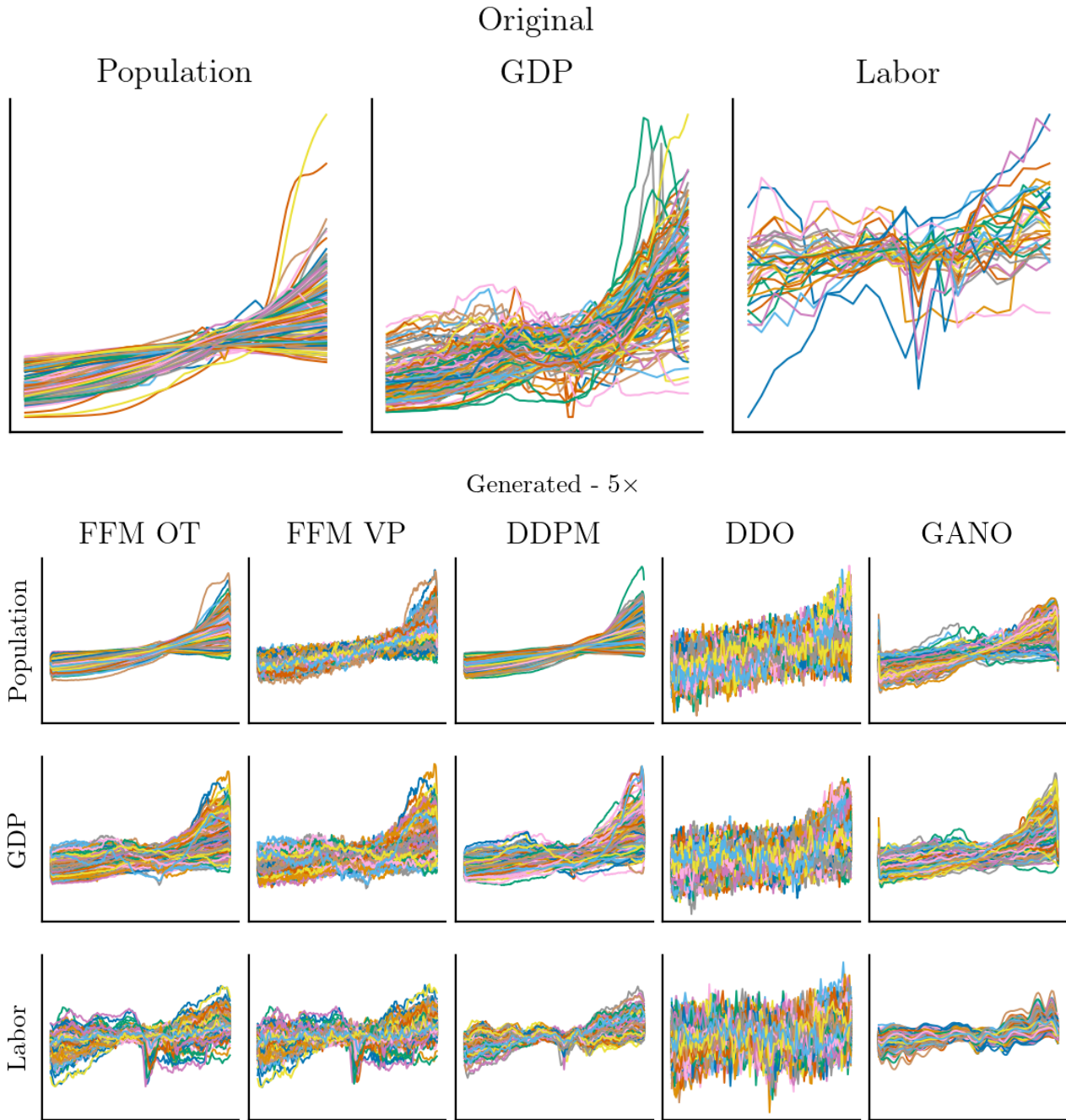


Figure 8: Samples from the three economics datasets and samples from the various models at a 5x super-resolution.

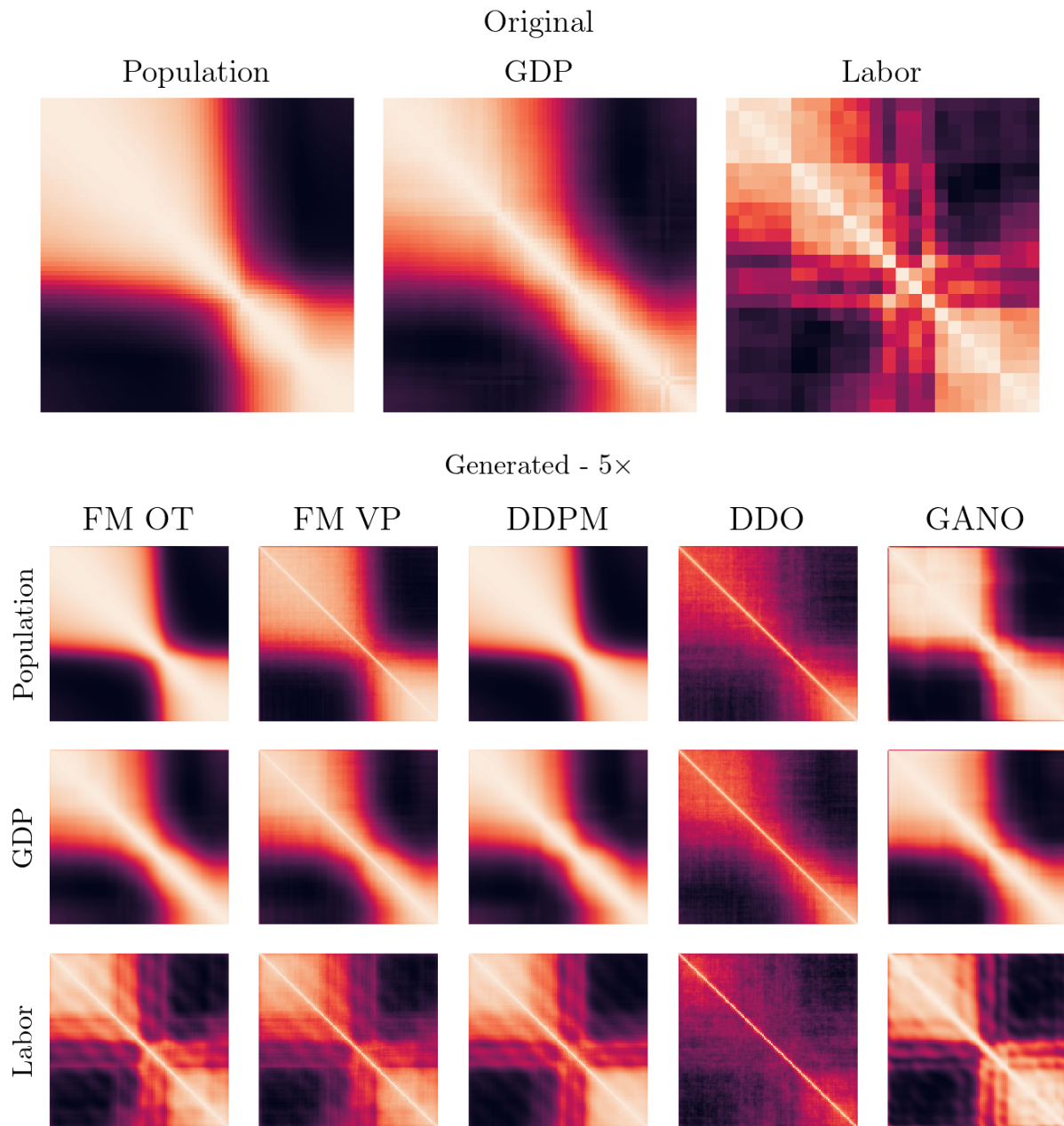


Figure 9: Correlation matrices for the three economics datasets (top row), as well as correlation matrices for each dataset for the generated samples from each model at a 5x super-resolution.

A.3.3 Additional Results: 1D Datasets

This section provides a further analysis and visualization of the generated samples for the 1D datasets considered in the main paper. In Figures 10 through 14, we plot the original data, generated samples from each model, and various pointwise statistics for the real and generated samples. Curves in black indicate the corresponding pointwise statistic for the original dataset (top row). The green error bands represent the minimal and maximal value of the pointwise statistic from 500 samples across ten random seeds of the corresponding model. The dashed green lines indicate the mean pointwise statistic across these ten runs for each model. See Table 1 for a quantitative comparison derived from these figures.

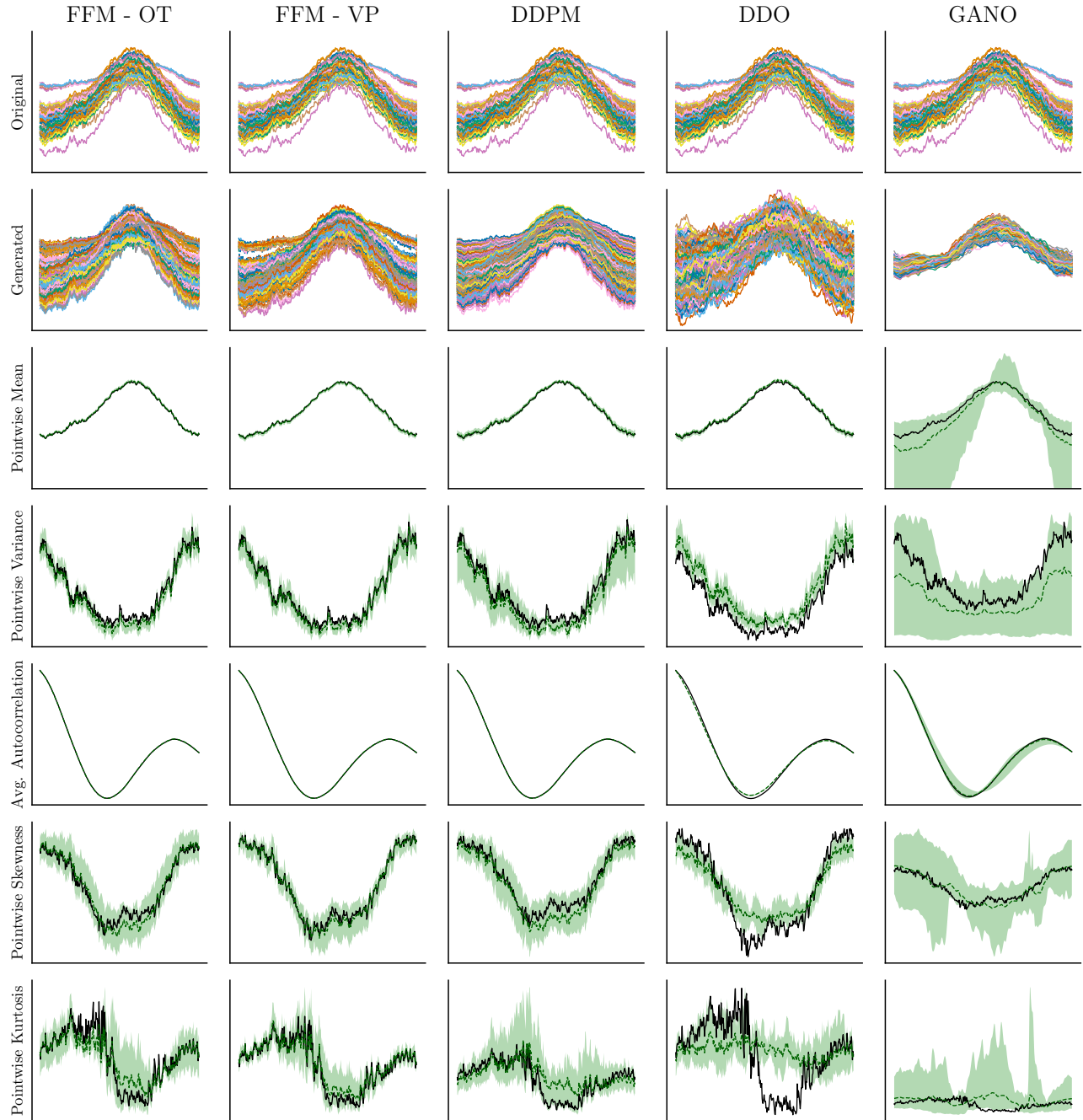


Figure 10: Various pointwise statistics on the AEMET dataset.

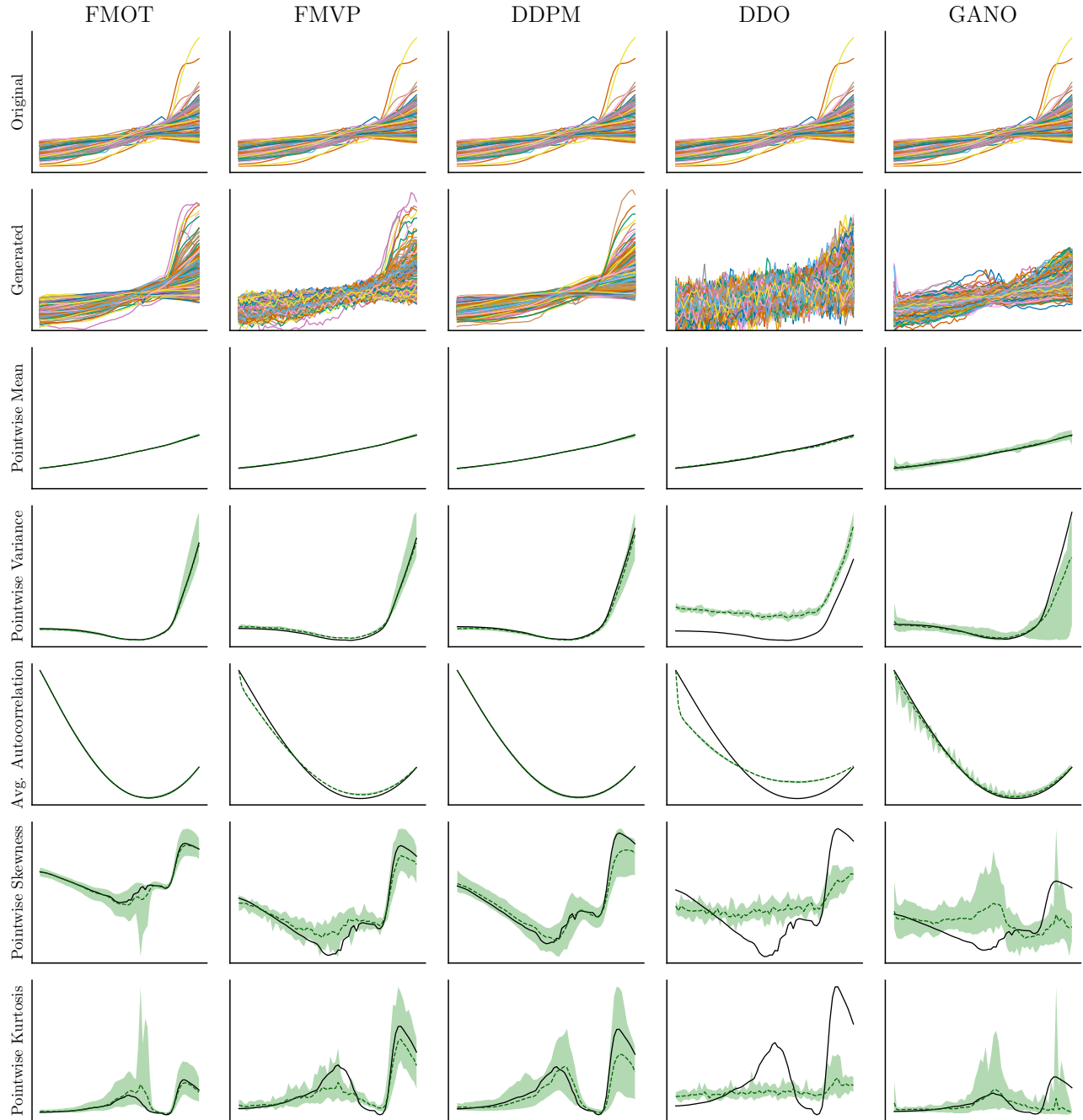


Figure 11: Various pointwise statistics on the Population dataset.

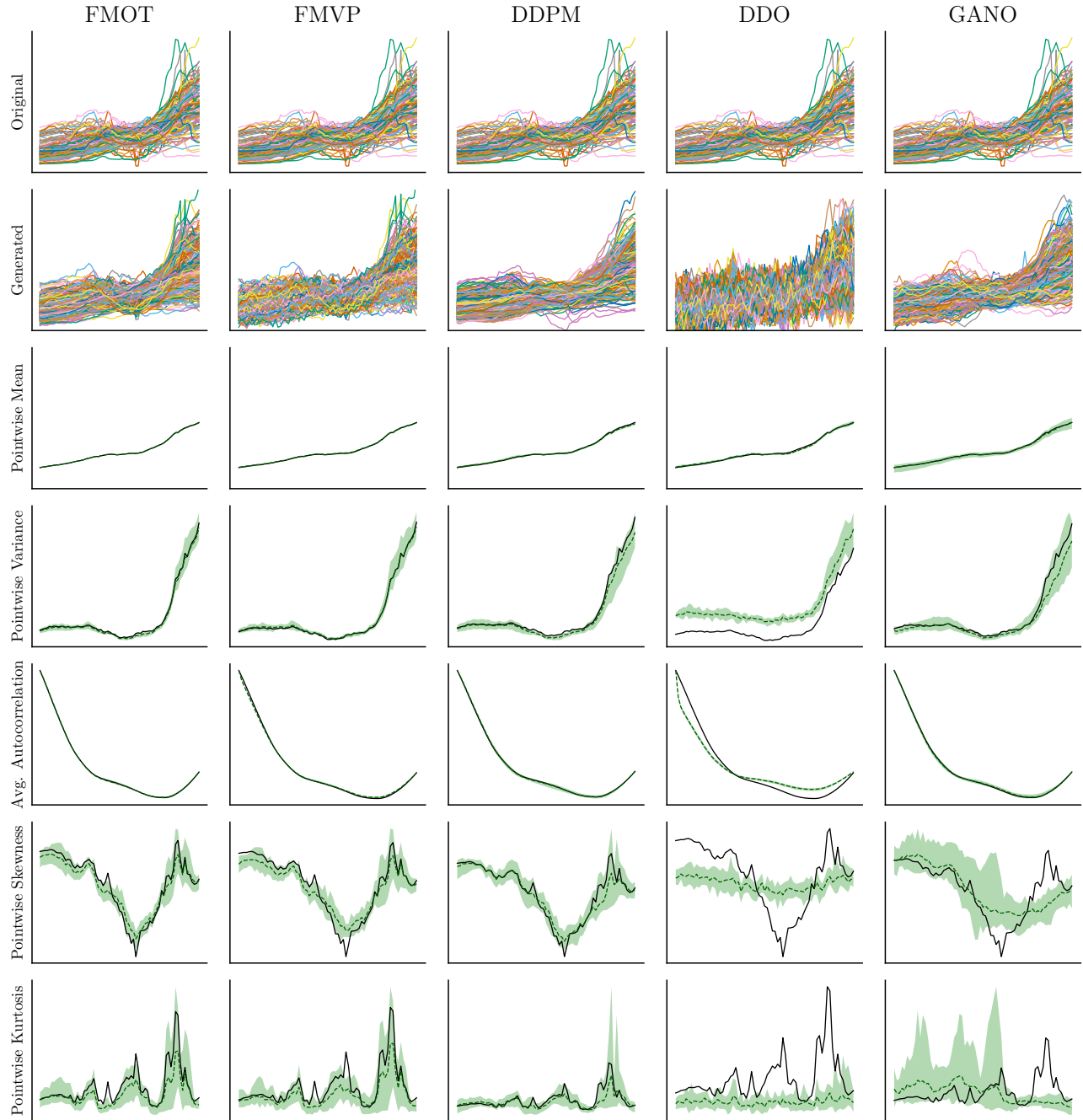


Figure 12: Various pointwise statistics on the GDP dataset.

Functional Flow Matching



Figure 13: Various pointwise statistics on the Labor dataset.

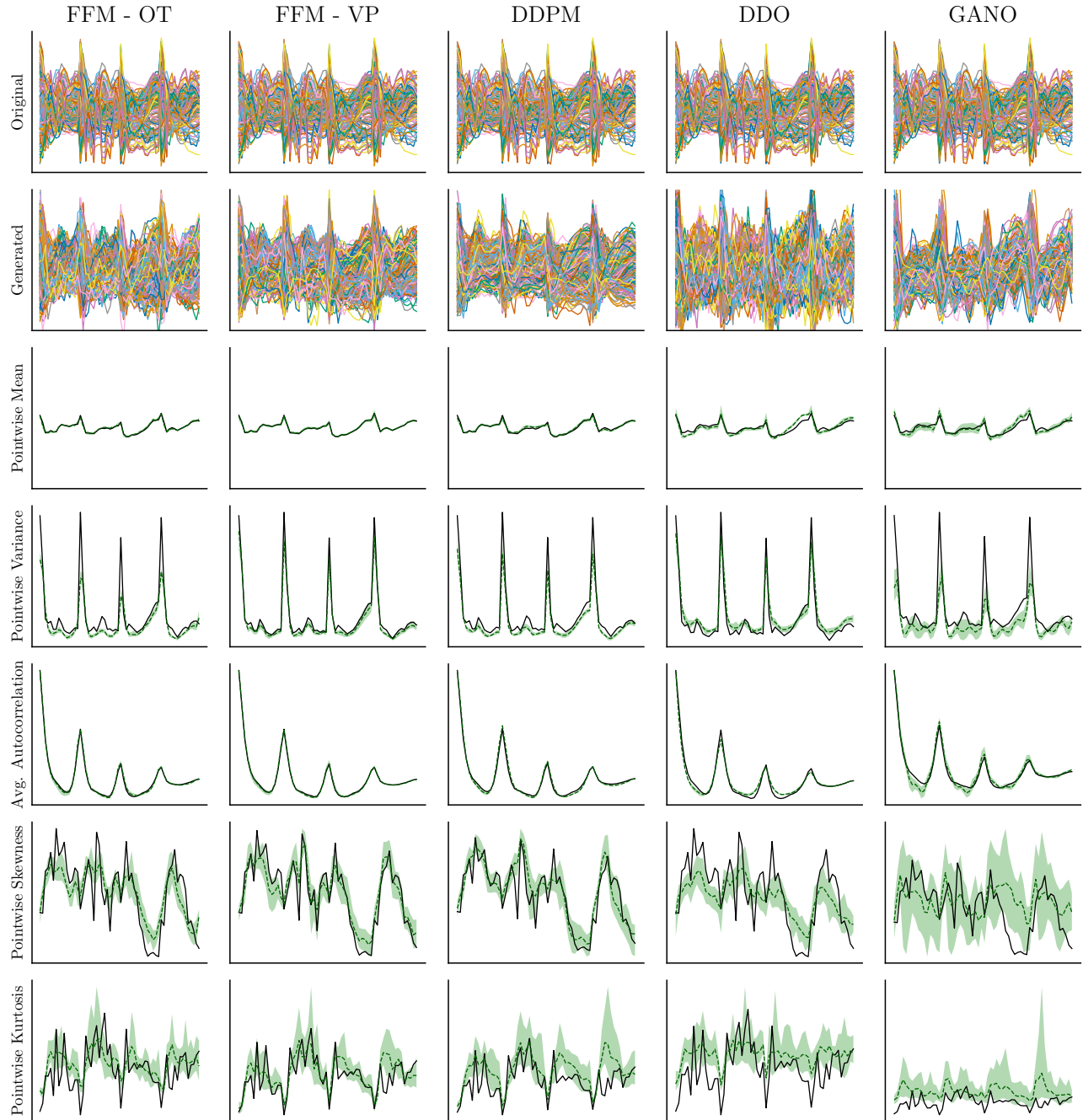


Figure 14: Various pointwise statistics on the Genes dataset.

A.3.4 Additional Results: Navier-Stokes Dataset

In this section, we provide additional visualizations and evaluation on the Navier-Stokes dataset, corresponding to the samples in Figure 2 and Table 2a in the main paper. The first row of Figure 15 plots a Gaussian KDE with a fixed bandwidth of 0.5 for the pixel-wise values of both the real and generated samples across all methods. We observe that FFM and DDPM closely match the ground-truth distribution, whereas DDO places too much mass around zero, and GANO learns a multimodal distribution. In the second row, we plot the spectrum of both the real and generated samples, i.e. the log-energy as a function of the wavenumber. We see that FFM and DDPM closely match the true spectrum for low wavenumbers, whereas the fits of DDO and GANO are less close. For all models, the generated samples fail to match the true spectrum at high wavenumbers. We obtain quantitative metrics from these visualizations by considering the pointwise MSE between the ground truth and generated curves to obtain the metrics in Table 2a.

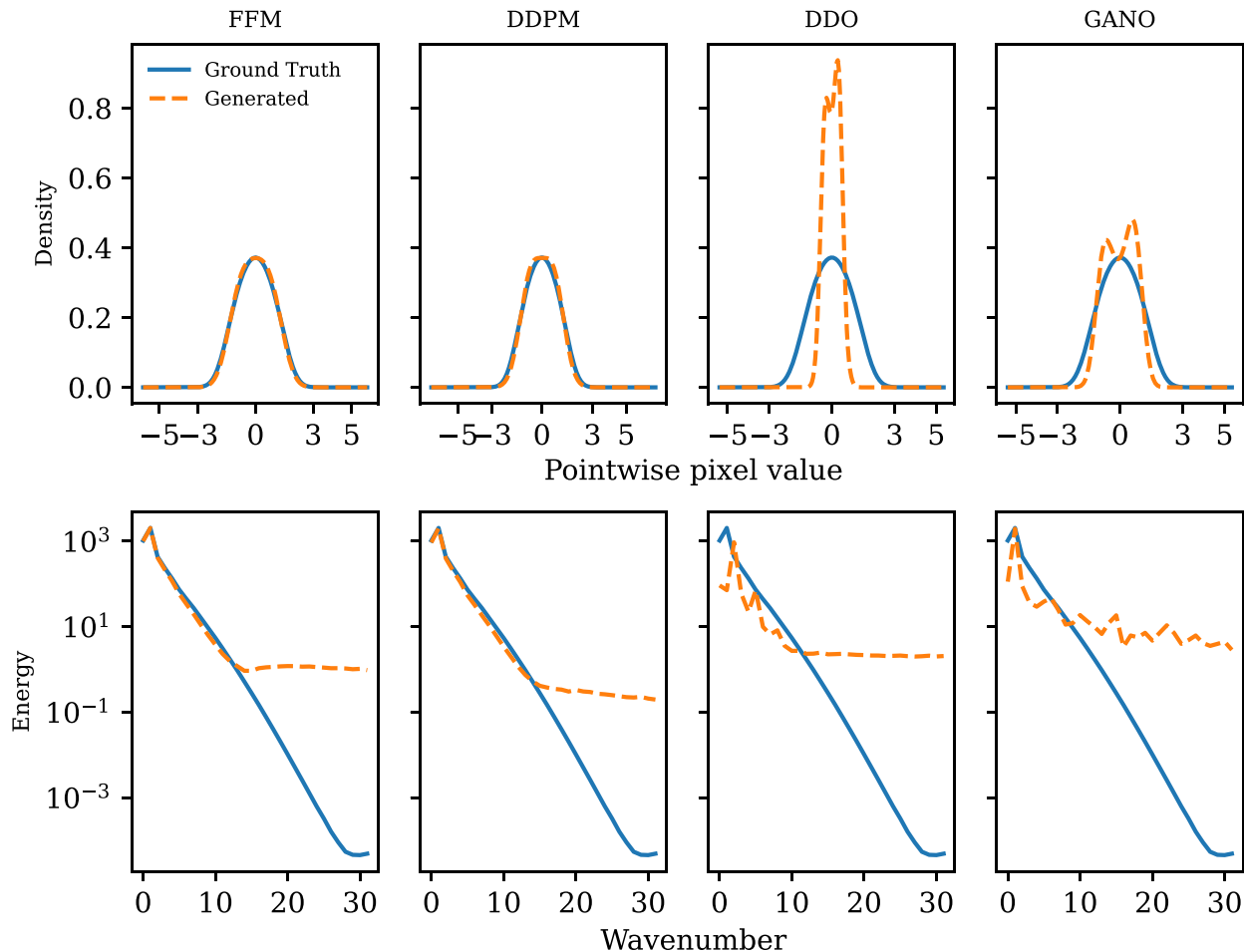


Figure 15: Additional visualizations corresponding to the samples in Figure 2 and Table 2a. We use 1000 samples from each of the models.

A.4 Conditional Models

In addition to unconditional generation of functions, we demonstrate that our method can be extended to perform conditional generation. That is, we have access to side information $z \in \mathbb{R}^d$ (assumed to be finite dimensional) and we are interested in sampling from the conditional data measure $\nu(df | z)$. For instance, z could be a collection of observed values of some function, and we may be interested in generating functions which interpolate (or extrapolate) these given observations. We describe two approaches to performing conditional generation: one based on a modified training process, and a second based on a modified sampling process. In Figure 5, we

demonstrate these two approaches on the AEMET dataset.

Conditional Training. Using the unconditional paths of measures μ_t^f as described in the unconditional setting, we may define a conditional marginal $\mu_t(\mathrm{d}f \mid z)$ by mixing over $\mathrm{d}\nu(f \mid z)$, i.e. $\mu_t(A \mid z) := \int_{\mathcal{F}} \mu_t^f(A) \mathrm{d}\nu(f \mid z)$.

As long as μ_t^f is concentrated around f , then $\mu_t(\mathrm{d}f \mid z) \approx \nu(\mathrm{d}f \mid z)$. Note that this condition is satisfied for the paths of measures constructed for unconditional generation, and hence no modification is necessary. However, modifying the conditional measures to account for the information z could potentially be beneficial, and we leave exploration of such design choices to future work. In all, we obtain a modified loss function

$$\mathcal{J}_C(\theta) = \mathbb{E}_{t \sim \mathcal{U}[0,1], z \sim q(z), f \sim \nu(\mathrm{d}f \mid z), g \sim \mu_t^f} \left[\left\| v_t^f(g) - u_t(f \mid z, \theta) \right\|^2 \right]. \quad (32)$$

In other words, we simply adapt our model architecture to also take in conditioning information z at training time. In practice, because z is assumed to be finite dimensional, we concatenate z to the input of our FNO model [Li et al., 2021]. We note that a similar loss appears in the context of Flow Matching generative models for video, as proposed by [Davtyan et al., 2022].

Conditional Sampling. As an alternative, we may instead modify the sampling process to account for z . This allows one to train an unconditional model and sample conditionally at generation time (in contrast to the conditional training setup, which only allows you to condition on the particular form of z you have trained on). Here, we assume that $z = (\vec{x}, \vec{y})$ consists of a collection of function observations, and that we would like to generate functions whose values match those observed in z .

In order to achieve this, at time $t \in [0, 1]$, we take a step as dictated by our ODE solver and model vector field to obtain a function \tilde{f}_t . Next, we flow the information contained in z forwards for t seconds along the conditional vector field designated by our model to obtain $z_t = (\vec{x}_t, \vec{y}_t)$. Then, we set $f_t(\vec{x}) = \vec{y}_t$. This approach can be seen as an extension of the ILVR method, which has been successfully applied to diffusion models for conditional image generation [Choi et al., 2021] and diffusion models for conditional function generation [Kerrigan et al., 2023].