Parameter estimation with data-driven nonparametric likelihood functions∗

Shixiao W. Jiang†
Department of Mathematics
The Pennsylvania State University

John Harlim‡
Department of Mathematics and Department of Meteorology
The Pennsylvania State University

April 11, 2018

Abstract

This paper presents a nonparametric modeling technique to avoid expensive evaluation of the likelihood function in estimating parameters of dynamical systems. The proposed method trains the likelihood function using the spectral expansion formulation known as the kernel embedding of conditional distribution. In particular, the spectral expansion uses a set of basis functions of a weighted Hilbert space defined on the data manifold with sampling density as the weight function. This choice of basis functions is justified by a rigorous error estimate. Numerically, these basis functions are estimated using the diffusion maps algorithm with variable bandwidth kernels on the training data set. The estimated likelihood function and posterior densities of the parameters based on this data-driven basis expansion will be compared to those that use analytical basis functions such as the Hermite polynomials and the Fourier cosine functions, which enforce simple geometry assumptions on the data manifold. The numerical results demonstrate the robustness of the data-driven basis functions on parameter estimation of stochastic and deterministic differential equations. In particular, the data-driven basis expansion produces superior estimates compared to those from the Hermite and cosine basis expansions when the intrinsic dimension of the data manifold is strictly less than the dimension of the ambient space.

1 Introduction

Bayesian inference is a popular approach for solving inverse problems with far reaching applications, such as parameter estimation and uncertainty quantification (see for example [14, 25, 8]). In this article, we will focus on a classical Bayesian inference problem of estimating the conditional distribution of hidden parameters of dynamical systems from a given set of noisy observations. In particular, let \( x(t; \theta) \) be a time dependent state variable, which implicitly depends on the parameter \( \theta \) through the following initial value problem,

\[
\dot{x} = f(x, \theta), \quad x(0) = x_0.
\]
Here, for any fixed $\theta$, $f$ can be either deterministic or stochastic. Our goal is to estimate the conditional distribution of $\theta$, given discrete-time noisy observations $y^t = \{y^t_1, \ldots, y^t_T\}$,

$$y^t_i = h(x^t_i) + \xi_i, \quad i = 1, \ldots, T,$$

where $x^t_i \equiv x(t_i; \theta^t)$ are the solutions of Eq. (1) for a specific hidden parameter $\theta^t$, $h$ is the observation function, and $\xi_i$ are unbiased and i.i.d. noises representing the measurement error. Although the proposed approach can also estimate the conditional density of the initial condition $x_0$, we will not explore this inference problem in this article. In our setup, we assume that the functional form of $f$ in (1) and $h$ in (2), the distribution of the noises $\xi_i$, and the initial condition $x_0$ are known.

Given a prior density, $p_0(\theta)$, Bayes’ theorem states that the conditional distribution of the parameter $\theta$ can be estimated as,

$$p(\theta|y^t) \propto p(y^t|\theta)p_0(\theta),$$

where $p(y^t|\theta)$ denotes the likelihood function of $\theta$ given the measurements $y^t$ that depend on a hidden parameter value $\theta^t$ through (2). In most applications, the statistics of the conditional distribution $p(\theta|y^t)$ are the quantity of interest. For example, one can use the mean statistic as a point estimator of $\theta$ parameter value and the higher order moments for uncertainty quantification. To realize this goal, one draws samples of $p(\theta|y^t)$ and estimates these statistics via Monte Carlo averages over these samples. In this application, Markov Chain Monte Carlo (MCMC) is a natural sampling method that plays a central role in the computational statistics behind most Bayesian inference techniques [6].

One way to define the likelihood function of $\theta$, $p(y^t|\theta)$, is as a product of the density functions of the i.i.d. measurement noises $\xi_i$:

$$p(y^t|\theta) = \prod_{i=1}^T p(\xi_i) = \prod_{i=1}^T p(y^t_i - h(x^t_i(\theta))),$$

where $x^t_i(\theta) \equiv x^t_i(t_i; \theta)$ are the solutions of the initial value problem in Eq. (1). Although this parametric modeling approach is convenient, in general, the dependence of the likelihood function on the parameter is implicit through the solutions $x^t_i(\theta)$. Practically, this implicit dependence will increase the computational cost in evaluating the likelihood function since it requires solving the dynamical model in (1) at every proposal in the MCMC chain.

Broadly speaking, the existing approaches to overcome this practical issue can be grouped into two classes. The first class consists of methods that improve/accelerate the sampling strategy; for example, the Hamiltonian Monte Carlo [21], adaptive MCMC [2], delay rejection adaptive Metropolis [9], just to name a few. The second class consists of methods that avoid solving the dynamical model in (1) when running the MCMC chain by replacing it with a computationally more efficient model. This class of approach, also known as surrogate modeling, includes Gaussian process models [12], polynomial chaos [19, 18], and enhanced model error [13]. Another related approach, which also avoids MCMC on top of integrating (1), is to employ a polynomial expansion on the likelihood function [20]. This method represents the parametric likelihood function in (4) with orthonormal basis functions of a Hilbert space weighted by the prior measure. This choice of basis functions makes the computation for the statistics of the posterior density straightforward and thus MCMC is not needed.

In this work, the proposed method is an extension of the surrogate modeling approach where MCMC is employed with a nonparametric likelihood function constructed using data-driven spectral expansion. By nonparametric, we mean that our approach does not model the likelihood function as a product of the given densities of the random noises $\xi_i$ as in (4). Instead, we approximate the likelihood function using the kernel embedding of conditional distribution formulation introduced in [24, 23]. In our application, we will extend their formulation onto a Hilbert space weighted by the sampling measure of the training dataset as in [5]. We will rigorously demonstrate that using the orthonormal basis functions of this data-driven weighted Hilbert
space, the number of required training data can be controlled for accurate likelihood function estimations. Computationally, given observation data that lie on a Riemannian manifold \( N \) embedded in \( \mathbb{R}^n \) with sampling density \( q(y) \), we apply the diffusion maps algorithm \([7,4]\) to approximate these basis functions using the training dataset. In this fashion, our approach respects the geometry of the data manifold since it represents the likelihood function using basis functions of \( L^2(N,q) \) with the sampling density \( q \) defined with respect to the volume form inherited by \( N \) from \( \mathbb{R}^n \). Using this nonparametric likelihood function, we then generate the MCMC chain for estimating the conditional distribution of hidden parameters.

This paper is organized as follows: In Sec. 2, we review the formulation of the reproducing kernel Hilbert space to estimate conditional density functions. In Sec. 3, we discuss the error estimate of the likelihood function approximation. In Sec. 4, we discuss the construction of the analytic basis functions for Euclidean space as well as the data-driven basis functions with diffusion maps algorithm for data lie on embedded Riemannian geometry. In Sec. 5, we provide numerical results with parameter estimation application on data manifold as well as the data-driven basis functions with diffusion maps algorithm for data lie on embedded Riemannian geometry. In Sec. 6, we conclude this paper with a short summary. We accompany this paper with a supplementary material which presents some details about calculations and discussions of numerical examples.

## 2 Conditional density estimation via reproducing kernel weighted Hilbert spaces

In our problem, we assume that the observation \( y \in N \subseteq \mathbb{R}^n \), where \( N \) is a smooth manifold with intrinsic dimension \( d \leq n \). In practice, we measure the observations in the ambient coordinates and denote its component as \( y = \{y_1, \ldots, y_n\} \). For the parameter \( \theta \) space, \( M \) has an Euclidean structure with components, \( \theta = \{\theta_1, \ldots, \theta_m\} \), so \( M \) is assumed to be either an \( m \)-dimensional hyperrectangle or \( \mathbb{R}^m \). For training, we are given \( M \) number of training parameters \( \{\theta_j\}_{j=1}^M = \{\theta_{j_1}, \ldots, \theta_{j_m}\}_{j=1}^M \). For each training parameter \( \theta_j \), we generate a discrete time series of length \( N \) for noisy observation data \( \{y_{i,j}\} = \{y_{i,j}^1, \ldots, y_{i,j}^N\} \in \mathbb{R}^n \) for \( i = 1, \ldots, N \), and \( j = 1, \ldots, M \). Here, the sub-index \( i \) and the sub-index \( j \) of \( y_{i,j} \) correspond to the \( i \)th observation data for the \( j \)th training parameter \( \theta_j \). Our goal for training is to learn the conditional density \( p(y|\theta) \) for arbitrary \( y \) and \( \theta \) from the training dataset \( \{\theta_j\}_{j=1,\ldots,M} \) and \( \{y_{i,j}\}_{j=1,\ldots,M} \).

The construction of the conditional density \( p(y|\theta) \) is based on a machine learning tool known as the kernel embedding of conditional distributions formulation introduced in \([24, 23]\). In their formulation, the representation of conditional distributions is an element of a reproducing kernel Hilbert space (RKHS). Recently, we generalized the representation to an element of a reproducing kernel weighted Hilbert space (RKWHS) in \([5]\). In this work, we will use our generalized formulation in \([5]\) for training to represent conditional density functions based on RKWHS. The outcome of the training is an estimate of the conditional density, \( \hat{p}(y|\theta) \), for arbitrary \( y \) and \( \theta \).

### 2.1 Review of nonparametric RKWHS representation of conditional density functions

We first review the RKWHS representation of conditional density functions deduced in \([5]\). Let \( \psi_k(y) \) be the orthonormal basis functions of \( L^2(N,q) \), where \( N \) contains the domain of the training data \( y_{i,j} \), and the weight function \( q(y) \) is defined with respect to the volume form inherited by \( N \) from the ambient space \( \mathbb{R}^n \). Let \( \varphi_l(\theta) \in L^2(M, \bar{q}) \) be the orthonormal basis functions in the parameter \( \theta \) space, where the training parameters are \( \theta_j \in M \), with weight function \( \bar{q}(\theta) \). For finite modes, \( k = 1, \ldots, K_1 \), and \( l = 1, \ldots, K_2 \), a
nonparametric RKWHS representation of the conditional density can be written as follows [5]:

\[ \hat{p}(y|\theta) = \sum_{k=1}^{K_1} \hat{c}_{Y|\theta,k} \psi_k(y) q(y), \]

(5)

where \( \hat{p}(y|\theta) \) denotes an estimate of the conditional density \( p(y|\theta) \), and the expansion coefficients are defined as

\[ \hat{c}_{Y|\theta,k} = \sum_{l=1}^{K_2} [C_{\Theta\Theta}C_{\Theta\Theta}^{-1}]_{kl} \varphi_l(\theta). \]

(6)

Here, the matrix \( C_{\Theta\Theta} \) is \( K_1 \times K_2 \) and the matrix \( C_{\Theta\Theta}^{-1} \) is \( K_2 \times K_2 \), whose components can be approximated by Monte Carlo averages [5]:

\[ [C_{\Theta\Theta}]_{ls} = E_{\Theta\Theta}[\varphi_s \varphi_l] \approx \frac{1}{MN} \sum_{j=1}^{M} \sum_{i=1}^{N} \psi_k(y_{i,j}) \varphi_s(\theta_j), \]

(7)

\[ [C_{\Theta\Theta}]_{sl} = E_{\Theta\Theta}[\varphi_s \varphi_l] \approx \frac{1}{M} \sum_{j=1}^{M} \varphi_s(\theta_j) \varphi_l(\theta_j), \]

(8)

where the expectations \( E \) are taken with respect to the sampling densities of the training dataset \( \{y_{i,j}\}_{j=1,...,M} \) and \( \{\theta_j\}_{j=1,...,M} \). The equation for the expansion coefficients in Eq. (6) is based on the theory of kernel embedding of conditional distribution [24, 23, 5]. See [5] for the detailed proof of Eqs. (6)-(8). Note that for RKWHS representation, the weight functions \( q \) and \( \bar{q} \) can be different from the sampling densities of the training dataset \( \{y_{i,j}\}_{i=1,...,N} \) and \( \{\theta_j\}_{j=1,...,M} \), respectively. This generalizes the representation in [5] which sets the weights \( q \) and \( \bar{q} \) to be the sampling densities of the training dataset \( \{y_{i,j}\} \) and \( \{\theta_j\} \), respectively.

Incidentally, it is worth mentioning that the conditional density in (5) and (6) is represented as a regression in infinite-dimensional spaces with basis functions \( \psi_k(y) \) and \( \varphi_l(\theta) \). The expression (5) is a nonparametric representation in the sense that we do not assume any particular distribution for the density function \( p(y|\theta) \). In this representation, only the training dataset \( \{y_{i,j}\}_{i=1,...,N} \) and \( \{\theta_j\}_{j=1,...,M} \) with appropriate basis functions are used to specify the coefficients \( \hat{c}_{Y|\theta,k} \) and the densities \( \hat{p}(y|\theta) \). In Sec. 4, we will demonstrate how to construct the appropriate basis completely from the training data, motivated by the theoretical result in Sec. 3 below.

### 2.2 Simplification of the expansion coefficients (6)

If the weight function \( \bar{q}(\theta) \) is the sampling density of the training parameters \( \{\theta_j\}_{j=1,...,M} \), the matrix \( C_{\Theta\Theta}^{-1} \) in (8) can be simplified to be a \( K_2 \times K_2 \) identity matrix,

\[ [C_{\Theta\Theta}]_{sl} = E_{\Theta\Theta}[\varphi_s \varphi_l] = \int_{\mathcal{M}} \varphi_s(\theta) \varphi_l(\theta) \bar{q}(\theta) d\theta = \delta_{sl}, \]

(9)

where \( \delta_{sl} \) is the kronecker delta function. Here, the second equality follows from the weight \( \bar{q}(\theta) \) being the sampling density, and the third equality follows from the orthonormality of \( \varphi_l(\theta) \in L^2(\mathcal{M}, \bar{q}) \) with respect to the weight function \( \bar{q} \). Then the expansion coefficients \( \hat{c}_{Y|\theta,k} \) in (6) can be simplified as

\[ \hat{c}_{Y|\theta,k} = \sum_{l=1}^{K_2} [C_{\Theta\Theta}]_{kl} \varphi_l(\theta), \]

(10)

with the \( K_1 \times K_2 \) matrix \( C_{\Theta\Theta} \) is still given by (7). In this work, we always take the weight function \( \bar{q}(\theta) \) to be the sampling density of the training parameters \( \{\theta_j\}_{j=1,...,M} \) for the simplification of the expansion
coefficients $\hat{c}_{Y|\Theta,k}$ in (10). This assumption is not too restrictive since the training parameters are specified by the users.

Finally, the formula in (5) combined with the expansion coefficients $\hat{c}_{Y|\Theta,k}$ in (10) and the matrix $C_{Y\Theta}$ in (7) forms a RKWH representation of the conditional density $p(y|\Theta)$ for arbitrary $y$ and $\Theta$. Numerically, the training outcome is the matrix $C_{Y\Theta}$ in (7) and then the conditional density $\hat{p}(y|\Theta)$ can be represented by (5) with coefficients (10) using the basis functions $\{\psi_k(y)\}_{k=1}^{K_2}$ and $\{\varphi_l(\Theta)\}_{l=1}^{K_2}$.

From above, one can see that two important questions naturally arise as a consequence of the usage of RKWH representation. First, whether the representation $\hat{p}(y|\Theta)$ in (5) is valid in estimating the conditional density $p(y|\Theta)$. Second, how to construct the orthonormal basis functions $\psi_k(y) \in L^2(\mathcal{N},q)$ and $\varphi_l(\Theta) \in L^2(\mathcal{M},q)$. We will address these two important questions in the next two sections.

### 3 Error estimation

In this section, we focus on the error estimation of the expansion coefficient $\hat{c}_{Y|\Theta,k}$, and later the conditional density $\hat{p}(y|\Theta)$ at the training parameter $\Theta$. The notation $\hat{c}_{Y|\Theta,k}$ is defined as the expansion coefficient $\hat{c}_{Y|\Theta,k}$ in (10), evaluated at the training parameter $\Theta$. Let the total number of basis functions in parameter space, $K_2$, be equal to the total number of training parameters, $M$, that is, $K_2 = M$, we can obtain the following relation,

$$
\frac{1}{M} \sum_{i=1}^{M} \varphi_i(\Theta_i) \varphi_i(\Theta_j) = \delta_{ij}.
$$

(11)

For the training parameter $\Theta_j$, we can simplify the expansion coefficient $\hat{c}_{Y|\Theta,j}$ by substituting Eq. (7) into Eq. (10),

$$
\hat{c}_{Y|\Theta,j} = \sum_{l=1}^{M} [C_{Y\Theta}]_{jl} \varphi_l(\Theta_j)
\approx \sum_{l=1}^{M} \left[ \frac{1}{MN} \sum_{s=1}^{M} \sum_{i=1}^{N} \psi_k(y_{i,s}) \varphi_l(\Theta_s) \right] \varphi_l(\Theta_j)\frac{1}{N} \sum_{i=1}^{N} \psi_k(y_{i,j}),
$$

(12)

where the last equality follows from (11).

#### 3.1 Error estimation using arbitrary bases

We first study the error estimation for the expansion coefficient $\hat{c}_{Y|\Theta,j,k}$. For each training parameter $\Theta_j$, the conditional density function $p(y|\Theta_j) \in L^2(\mathcal{N},q^{-1})$ can be analytically represented in the form,

$$
p(y|\Theta_j) = \sum_{k=1}^{\infty} c_{Y|\Theta,j,k} \psi_k(y) q(y),
$$

(13)

due to the completeness of $L^2(\mathcal{N},q)$. Here, the analytic expansion coefficient $c_{Y|\Theta,j,k}$ is given by,

$$
c_{Y|\Theta,j,k} = \langle p(\cdot|\Theta_j), \psi_k \rangle.
$$

(14)

Note that the estimator $\hat{c}_{Y|\Theta,j,k}$ in (12) is a Monte Carlo approximation of the expansion coefficient $c_{Y|\Theta,j,k}$ in (14), i.e.,

$$
c_{Y|\Theta,j,k} = \langle p(y|\Theta_j), \psi_k(y) \rangle = E_{Y|\Theta,j}[\psi_k(Y)] \approx \frac{1}{N} \sum_{i=1}^{N} \psi_k(y_{i,j}),
$$

(15)
where the last equality follows from the training dataset \( \{ y_{i,j} \}_{i=1,\ldots,N} \) which admits a conditional density \( p(y|\theta_j) \). Then we can obtain the following unbiasedness and consistency of the estimator \( \hat{c}_{Y|\theta_j,k} \).

**Proposition 1** Let \( \{ y_{i,j} \}_{i=1,\ldots,N} \) be i.i.d. samples of \( Y|\theta_j \) with density \( p(y|\theta_j) \). Let \( p(y|\theta_j) \in L^2(N,q^{-1}) \) and \( \{ \psi_k(y) \} \) form a complete orthonormal basis of \( L^2(N,q) \). Assume that \( \text{Var}_{\theta_j} [\psi_k(Y)] \) is finite, then \( \hat{c}_{Y|\theta_j,k} \) defined in (12) is an unbiased and consistent estimator for \( c_{Y|\theta_j,k} \) in (14).

**Proof** The estimator \( \hat{c}_{Y|\theta_j,k} \) is unbiased,

\[
E \hat{c}_{Y|\theta_j,k} = \frac{1}{N} \sum_{i=1}^{N} E_{Y|\theta_j} \psi_k(Y_{i,j}) = c_{Y|\theta_j,k}.
\]

where the expectation is taken with respect to the conditional density \( p(y|\theta_j) \). If the variance, \( \text{Var}_{\theta_j} [\psi_k(Y)] \), is finite, then the variance of \( \hat{c}_{Y|\theta_j,k} \) converges to 0 as the number of training data \( N \to \infty \),

\[
\text{Var}(\hat{c}_{Y|\theta_j,k}) = \frac{1}{N} \text{Var}_{\theta_j} [\psi_k(Y)] \to 0, \text{ as } N \to \infty.
\]

Then we can obtain that the estimator \( \hat{c}_{Y|\theta_j,k} \) is consistent,

\[
P_r \left( |\hat{c}_{Y|\theta_j,k} - c_{Y|\theta_j,k}| > \varepsilon \right) \leq \frac{\text{Var}(\hat{c}_{Y|\theta_j,k})}{\varepsilon^2} \to 0, \text{ as } N \to \infty, \text{ for } \forall \varepsilon > 0,
\]

where Chebyshev’s inequality has been used.

If the estimator of \( p(y|\theta_j) \) is given by the representation with infinite number of basis functions, \( \bar{p}(y|\theta_j) = \sum_{k=1}^{\infty} \hat{c}_{Y|\theta_j,k} \psi_k(y)q(y) \), then the estimator \( \bar{p}(y|\theta_j) \) is pointwise unbiased for every observation \( y \). However, in numerical implementation, only finite number of basis functions can be used in representation (5). Numerically, the estimator of \( p(y|\theta_j) \) is given by the representation (5) at the training parameter \( \theta_j \),

\[
\bar{p}(y|\theta_j) = \sum_{k=1}^{K_1} \hat{c}_{Y|\theta_j,k} \psi_k(y)q(y).
\]

Then the pointwise error of the estimator, \( \bar{e}(y|\theta_j) \), can be defined as

\[
\bar{e}(y|\theta_j) = p(y|\theta_j) - \bar{p}(y|\theta_j) = \sum_{k=K_1+1}^{\infty} c_{Y|\theta_j,k} \psi_k(y)q(y) + \sum_{k=1}^{K_1} \left( c_{Y|\theta_j,k} - \hat{c}_{Y|\theta_j,k} \right) \psi_k(y)q(y).
\]

It can be seen that the estimator \( \bar{p}(y|\theta_j) \) is no longer unbiased or consistent due to the first error term in (18) induced by modes \( k > K_1 \). Next, we estimate the expectation and the variance of a \( L^2 \)-norm error of \( \bar{p}(y|\theta_j) \) for all training parameters \( \theta_j \).

**Theorem 1** Let the condition in Proposition 1 be satisfied for all \( \{ \theta_j \}_{j=1,\ldots,M} \), and \( \text{Var}_{\theta_j} [\psi_k(Y)] \) be finite for all \( k \in \mathbb{N}^+ \). Define the \( L^2 \)-norm error,

\[
\bar{e}_{L^2} = \left( \sum_{j=1}^{M} \int_N |\bar{e}(y|\theta_j)|^2 q^{-1}(y)dV(y) \right)^{1/2},
\]
where \( \hat{e}(y|\theta_j) \) is the pointwise error in (18), and \( dV(y) \) is the volume form inherited by the manifold \( N \) from the ambient space \( \mathbb{R}^n \) \([4, 5]\). Then,

\[
E[\hat{e}^2] \leq \left( \sum_{j=1}^{M} \sum_{k=K_{1}+1}^{\infty} [c_{Y|\theta_j,k}]^2 + \frac{1}{N} \sum_{j=1}^{M} \sum_{k=1}^{K_{1}} \text{Var}_{Y|\psi_k(Y)} \right)^{\frac{1}{2}},
\]

(20)

\[
\text{Var}[\hat{e}^2] \leq \sum_{j=1}^{M} \sum_{k=K_{1}+1}^{\infty} [c_{Y|\theta_j,k}]^2 + \frac{1}{N} \sum_{j=1}^{M} \sum_{k=1}^{K_{1}} \text{Var}_{Y|\psi_k(Y)},
\]

(21)

where \( E \) and \( \text{Var} \) are defined with respect to joint distribution of \( p(y|\theta_j) \) for all \( \{\theta_j\}_{j=1,\ldots,M} \). Moreover, \( E[\hat{e}^2] \) and \( \text{Var}[\hat{e}^2] \) converge to zero as \( K_1 \to \infty \) and then \( N \to \infty \), where the limiting operations of \( K_1 \) and \( N \) are not commutative.

**Proof** The expectation of \( \hat{e}^2 \) can be estimated as,

\[
(E[\hat{e}^2])^2 \leq E \left[ \sum_{j=1}^{M} \int_{N} |\hat{e}(y|\theta_j)|^2 q^{-1}(y)dV(y) \right]
= E \left[ \sum_{j=1}^{M} \int_{N} \left( \sum_{k=K_{1}+1}^{\infty} c_{Y|\theta_j,k} \psi_k(y) + \sum_{k=1}^{K_{1}} (c_{Y|\theta_j,k} - \hat{c}_{Y|\theta_j,k}) \psi_k(y) \right)^2 q(y)dV(y) \right],
\]

(22)

where the first inequality follows from Jensen’s inequality. Here, the randomness comes from the estimators \( \hat{c}_{Y|\theta_j,k} \). Due to the orthonormality of basis functions \( \psi_k(y) \in L^2(N, q) \), the error estimation in (22) can be simplified as,

\[
(E[\hat{e}^2])^2 \leq \sum_{j=1}^{M} \sum_{k=K_{1}+1}^{\infty} [c_{Y|\theta_j,k}]^2 + \sum_{j=1}^{M} \sum_{k=1}^{K_{1}} E_{Y|\theta_j} \left[ (c_{Y|\theta_j,k} - \hat{c}_{Y|\theta_j,k})^2 \right],
\]

\[
= \sum_{j=1}^{M} \sum_{k=K_{1}+1}^{\infty} [c_{Y|\theta_j,k}]^2 + \frac{1}{N} \sum_{j=1}^{M} \sum_{k=1}^{K_{1}} \text{Var}_{Y|\theta_j} \left[ \psi_k(Y) \right],
\]

(23)

where the inequality follows from the linearity of expectation, and the equality follows from \( E\hat{c}_{Y|\theta_j,k} = c_{Y|\theta_j,k} \) in (16) and \( \text{Var}(\hat{c}_{Y|\theta_j,k}) = \frac{1}{N} \text{Var}_{Y|\theta_j} \left[ \psi_k(Y) \right] \) in (17). Note that in error estimation (23), the first term is deterministic and the second term is random. We have so far proved that the expectation \( E[\hat{e}^2] \) is bounded by (20). Similarly, we can prove that the variance \( \text{Var}[\hat{e}^2] \) is bounded by (21).

Next, we prove that the expectation \( E[\hat{e}^2] \) converges to zero as \( K_1 \to \infty \) and then \( N \to \infty \). Parseval’s theorem states that

\[
\sum_{k=1}^{\infty} [c_{Y|\theta_j,k}]^2 = \int_{N} p(y|\theta_j)^2 q^{-1}(y) dV(y) < +\infty, \text{ for all } \theta_j,
\]

(24)

where the inequality follows from \( p(y|\theta_j) \in L^2(N, q^{-1}) \) for all \( \theta_j \). For \( \forall \varepsilon > 0 \), there exists an integer \( K_1(\theta_j) \) for \( \theta_j \) such that

\[
\sum_{k=K_1(\theta_j)}^{\infty} [c_{Y|\theta_j,k}]^2 < \frac{\varepsilon}{2M}.
\]

(25)

Let

\[
K_1 = \max \left\{ K_1(\theta_1), \ldots, K_1(\theta_M) \right\},
\]

(26)
then the first term in (23) can be bounded by $\varepsilon/2$,
\begin{equation}
\sum_{j=1}^{M} \sum_{k=K+1}^{\infty} \| \mathcal{C}_{\mathbf{Y} | \mathbf{\theta}_j, k} \|^2 < \frac{\varepsilon}{2}.
\end{equation}

Since the variance $\text{Var}_{\mathbf{Y} | \mathbf{\theta}_j} [\psi_k (\mathbf{Y})]$ is assumed to be finite for all $k$ and $j$, there exists a constant $D > 0$ such that $\text{Var}_{\mathbf{Y} | \mathbf{\theta}_j} [\psi_k (\mathbf{Y})]$ can be bounded above by this constant $D$,
\begin{equation}
\text{Var}_{\mathbf{Y} | \mathbf{\theta}_j} [\psi_k (\mathbf{Y})] \leq D, \quad \text{for all } k = 1, \ldots, K_1 \text{ and } j = 1, \ldots, M.
\end{equation}

Then for $\forall \varepsilon > 0$, there exists a sufficiently large number of training data
\begin{equation}
N_{\text{min}} = \frac{2MK_1D}{\varepsilon}.
\end{equation}

such that whenever $N > N_{\text{min}}$, then
\begin{equation}
\frac{1}{N} \sum_{j=1}^{M} \sum_{k=1}^{K_1} \text{Var}_{\mathbf{Y} | \mathbf{\theta}_j} [\psi_k (\mathbf{Y})] < \frac{\varepsilon}{2}.
\end{equation}

Since $\varepsilon > 0$ is arbitrary, by substituting Eq. (27) and Eq. (30) into the error estimation (23), we obtain that $\mathbb{E} [\tilde{e}_{L,2}]$ converges to zero as $K_1 \to \infty$ and then $N \to \infty$. Note that, we first take $K_1 \to \infty$ to ensure the first error term in (23) vanishes, and then take $N \to \infty$ to ensure the second error term in (23) vanishes. Thus, the limiting operations of $K_1$ and $N$ are not commutative. Similarly, we can prove that the variance $\text{Var}[\tilde{e}_{L,2}]$ converges to zero as $K_1 \to \infty$ and then $N \to \infty$.

Theorem 1 provides an intuition for specifying the number of training observation data $N$ to achieve any desired accuracy $\varepsilon > 0$ given fixed $M$-parameters and sufficiently large $K_1$. It can be seen from Theorem 1 that numerically, the expectation $\mathbb{E} [\tilde{e}_{L,2}]$ in (20) and the variance $\text{Var}[\tilde{e}_{L,2}]$ in (21) can be bounded within arbitrarily small $\varepsilon$ by choosing sufficiently large $K_1$ and $N$. Specifically, there are two error terms in Eqs. (20) and (21), the first being deterministic induced by modes $k > K_1$, and the second random induced by modes $k \leq K_1$. For the deterministic term ($k > K_1$), the error can be bounded by $\varepsilon/2$ by choosing sufficiently large $K_1$ satisfying (26). In our implementation, the number of basis functions $K_1$ is empirically chosen to be large enough in order to make the first error term in Eqs. (20) and (21) for $k > K_1$ as small as possible.

For the random term ($k \leq K_1$), the error can be bounded by $\varepsilon/2$ by choosing sufficiently large $N$ satisfying $N > N_{\text{min}} = 2MK_1D/\varepsilon$ [Eq. (29)]. The minimum number of training data, $N_{\text{min}}$, depends on the upper bound of $\text{Var}_{\mathbf{Y} | \mathbf{\theta}_j} [\psi_k (\mathbf{Y})]$, $D$. However, the upper bound $D$ may not exist for some problems. This means that for some problems, the assumption for finite $\text{Var}_{\mathbf{Y} | \mathbf{\theta}_j} [\psi_k (\mathbf{Y})]$ in Theorem 1 may be not satisfied. Even if the upper bound $D$ exists, it is typically not easy to evaluate its value given an arbitrary basis $\psi_k \in L^2 (\mathcal{N}, q)$ since one needs to evaluate $\text{Var}_{\mathbf{Y} | \mathbf{\theta}_j} [\psi_k (\mathbf{Y})]$ for all $k = 1, \ldots, K_1$ and $j = 1, \ldots, M$. Note that Theorem 1 holds true for representing $\hat{p}(\mathbf{y} | \mathbf{\theta}_j)$ with an arbitrary basis $\{ \psi_k \} \in L^2 (\mathcal{N}, q)$ as long as $p(\mathbf{y} | \mathbf{\theta}_j) \in L^2 (\mathcal{N}, q^{-1})$ for all $\mathbf{\theta}_j$ and $\text{Var}_{\mathbf{Y} | \mathbf{\theta}_j} [\psi_k (\mathbf{Y})]$ is finite for $k \leq K_1$ and $j = 1, \ldots, M$. Next, we provide several cases in which $\text{Var}_{\mathbf{Y} | \mathbf{\theta}_j} [\psi_k (\mathbf{Y})]$ is finite for $k$ and $j$.

**Remark 1** If the weighted Hilbert space $L^2 (\mathcal{N}, q)$ is defined on a compact manifold $\mathcal{N}$ and has smooth basis functions $\psi_k$, then $\text{Var}_{\mathbf{Y} | \mathbf{\theta}_j} [\psi_k (\mathbf{Y})]$ is finite for a fixed $k \in \mathbb{N}^+$ and $j = 1, \ldots, M$. This assertion follows from the fact that continuous functions on a compact manifold are bounded. The smoothness assumption is not unreasonable in many applications since the orthonormal basis functions are obtained as solutions of an eigenvalue problem of a self-adjoint second-order elliptic differential operator. Note that the bound here is not necessarily a uniform bound of $\psi_k (\mathbf{Y})$ for all $k \in \mathbb{N}^+$ and $j = 1, \ldots, M$. As long as $\text{Var}_{\mathbf{Y} | \mathbf{\theta}_j} [\psi_k (\mathbf{Y})]$ is finite for $k \leq K_1$ and $j = 1, \ldots, M$, the upper bound $D$ is finite and then Theorem 1 holds true.
Remark 2 If the manifold $\mathcal{N}$ is a hyperrectangle in $\mathbb{R}^n$ and the weight $q$ is a uniform distribution on $\mathcal{N}$, then $\text{Var}_{\mathbf{Y}\mid \theta_j} [\psi_k (\mathbf{Y})]$ is finite for a fixed $k \in \mathbb{N}^+$ and $j = 1, \ldots, M$. This assertion is an immediate consequence of Remark 1.

In Theorem 1, $N_{\text{min}}$ depends on the upper bound of $\text{Var}_{\mathbf{Y}\mid \theta_j} [\psi_k (\mathbf{Y})]$, $D$, as shown in (29). In the following, we will specify a Hilbert space, referred to as a data-driven Hilbert space, so that $N_{\text{min}}$ is independent of $D$ and is only dependent of $M$, $K_1$, and $\varepsilon$. As a consequence, we can easily determine how many training data $N$ for bounding the second error term in Eqs. (20) and (21).

3.2 Error estimation using a data-driven Hilbert space

We now turn to the discussion of a specific data-driven Hilbert space $L^2 (\mathcal{N}, \overline{q})$ with orthonormal basis functions $\overline{\psi}_k$. Our goal is to specify the weight function $\overline{q}$ such that the minimum number of training data, $N_{\text{min}}$, only depends on $M$, $K_1$, and $\varepsilon$. Here, the overline $\overline{\cdot}$ corresponds to the specific data-driven Hilbert space. The second error term in (23) can be further estimated as,

$$
\frac{1}{N} \sum_{j=1}^{M} \sum_{k=1}^{K_i} \text{Var}_{\mathbf{Y}\mid \theta_j} [\overline{\psi}_k (\mathbf{Y})] \leq \frac{1}{N} \sum_{k=1}^{K_i} \sum_{j=1}^{M} \mathbb{E}_{\mathbf{Y}\mid \theta_j} [\overline{\psi}_k^2 (\mathbf{Y})] \\
= \frac{M}{N} \sum_{k=1}^{K_i} \int_{\mathcal{N}} \overline{\psi}_k^2 (\mathbf{y}) \left( \frac{1}{M} \sum_{j=1}^{M} p(\mathbf{y}\mid \theta_j) \right) dV (\mathbf{y}), \quad (31)
$$

where the basis functions are substituted with the specific $\overline{\psi}_k$. Notice that $\overline{\psi}_k (\mathbf{y})$ are orthonormal basis functions with respect to the weight $\overline{q}$ in $L^2 (\mathcal{N}, \overline{q})$. One specific choice of the weight function $\overline{q} (\mathbf{y})$ is

$$
\overline{q} (\mathbf{y}) = \frac{1}{M} \sum_{j=1}^{M} p(\mathbf{y}\mid \theta_j), \quad (32)
$$

where $\overline{q} (\mathbf{y})$ has been normalized, i.e.,

$$
\int_{\mathcal{N}} \overline{q} (\mathbf{y}) dV (\mathbf{y}) = \int_{\mathcal{N}} \frac{1}{M} \sum_{j=1}^{M} p(\mathbf{y}\mid \theta_j) dV (\mathbf{y}) = 1. \quad (33)
$$

If the weight function $\overline{q} (\mathbf{y})$ is not normalized, there are only constant times for the error $\hat{c}_{L^2}$ in (19). Note that the weight function $\overline{q} (\mathbf{y})$ by (32) is a discretization of the marginal density function of $\mathbf{Y}$ with $\Theta$ marginalized out,

$$
\overline{q} (\mathbf{y}) = \frac{1}{M} \sum_{j=1}^{M} p(\mathbf{y}\mid \theta_j) \approx \int_{\mathcal{M}} p(\mathbf{y}\mid \theta) \overline{q} (\theta) d\theta = \int_{\mathcal{M}} p(\mathbf{y}\mid \theta) d\theta, \quad (34)
$$

where $p(\mathbf{y}, \Theta)$ denotes the joint density of $(\mathbf{Y}, \Theta)$. Essentially, the weight function $\overline{q} (\mathbf{y})$ in (32) is the sampling density of all the training data $\{\mathbf{y}_{i,j}\}_{j=1,\ldots,M}$, which motivates us to refer to $L^2 (\mathcal{N}, \overline{q})$ as a data-driven Hilbert space.

Next, we prove that by specifying the data-driven basis functions $\overline{\psi}_k \in L^2 (\mathcal{N}, \overline{q})$, the variance $\text{Var}_{\mathbf{Y}\mid \theta_j} [\overline{\psi}_k (\mathbf{Y})]$ is finite for all $k \in \mathbb{N}^+$ and $j = 1, \ldots, M$. Subsequently, we can obtain the minimum number of training data, $N_{\text{min}}$, to only depend on $M$, $K_1$, and $\varepsilon$, such that the expectation $\mathbb{E} [\hat{c}_{L^2}]$ in (20) and the variance $\text{Var} [\hat{c}_{L^2}]$ in (21) are bounded above by any $\varepsilon > 0$.

Proposition 2 Let $\{\mathbf{y}_{i,j}\}_{i=1,\ldots,N}$ be i.i.d. samples of $\mathbf{Y}\mid \theta_j$ with density $p(\mathbf{y}\mid \theta_j)$. Let $p(\mathbf{y}\mid \theta_j) \in L^2 (\mathcal{N}, \overline{q}^{-1})$ for all $\{\theta_j\}_{j=1,\ldots,M}$ with weight $\overline{q}$ specified in (32) and let $\{\overline{\psi}_k\}$ be the complete orthonormal basis of $L^2 (\mathcal{N}, \overline{q})$. Then $\text{Var}_{\mathbf{Y}\mid \theta_j} [\overline{\psi}_k (\mathbf{Y})]$ is finite for all $k \in \mathbb{N}^+$ and $j = 1, \ldots, M$. 

9
Proof Notice that for all $k \in \mathbb{N}^+$, we have

$$
\frac{1}{M} \sum_{j=1}^{M} \text{Var}_{Y|\theta_j} \left[ \overline{\psi}_k(Y) \right] \leq \frac{1}{M} \sum_{j=1}^{M} \mathbb{E}_{Y|\theta_j} \left[ \overline{\psi}_k^2(Y) \right] = \int_{\mathbb{N}} \overline{\psi}_k^2(y) \left( \frac{1}{M} \sum_{j=1}^{M} p(y|\theta_j) dV(y) \right) = \int_{\mathbb{N}} \overline{\psi}_k^2(y) \overline{\tau}(y) dV(y) = 1,
$$

(35)

where the last equality follows directly from the orthonormality of basis functions $\overline{\psi}_k(y) \in L^2(\mathbb{N}, \overline{\tau})$. From above Eq. (35), we can obtain that for all $k \in \mathbb{N}^+$ and $j = 1, \ldots, M$, the variance $\text{Var}_{Y|\theta_j} \left[ \overline{\psi}_k(Y) \right]$ is finite.

Theorem 2 Given the same hypothesis as in Proposition 2, then

$$
\mathbb{E} \left[ \hat{e}_{L^2} \right] \leq \left( \sum_{j=1}^{M} \sum_{k=K_1+1}^{\infty} \left[ c_{Y|\theta_j,k} \right]^2 + \frac{MK_1}{N} \right)^{1/2},
$$

(36)

$$
\text{Var} \left[ \hat{e}_{L^2} \right] \leq \sum_{j=1}^{M} \sum_{k=K_1+1}^{\infty} \left[ c_{Y|\theta_j,k} \right]^2 + \frac{MK_1}{N},
$$

(37)

where $\hat{e}_{L^2}$ is defined by (19) and $c_{Y|\theta_j,k}$ is given by (14). Moreover, $\mathbb{E} \left[ \hat{e}_{L^2} \right]$ and $\text{Var} \left[ \hat{e}_{L^2} \right]$ converge to zero as $K_1 \to \infty$ and then $N \to \infty$, where the limiting operations of $K_1$ and $N$ are not commutative.

Proof According to Proposition 2, we have that the variance $\text{Var}_{Y|\theta_j} \left[ \overline{\psi}_k(Y) \right]$ is finite for all $k \in \mathbb{N}^+$ and $j = 1, \ldots, M$. According to Proposition 1, since $\text{Var}_{Y|\theta_j} \left[ \overline{\psi}_k(Y) \right]$ is finite, we have that the estimator $\hat{e}_{Y|\theta_j,k}$ is both unbiased and consistent for $c_{Y|\theta_j,k}$. All conditions in Theorem 1 are satisfied, so that we can obtain the error estimation of the expectation $\mathbb{E} \left[ \hat{e}_{L^2} \right]$ in (20) and error estimation of the variance $\text{Var} \left[ \hat{e}_{L^2} \right]$ in (21). The second error term in $\mathbb{E} \left[ \hat{e}_{L^2} \right]$ (20) and $\text{Var} \left[ \hat{e}_{L^2} \right]$ (21) can be both bounded by Eq. (35), so that we can obtain our error estimations (36) and (37).

Choose $K_1$ as in (26) such that the first term in (36) and (37) is bounded by $\varepsilon/2$. The second term $MK_1/N$ in (36) and (37) can be bounded by an arbitrarily small $\varepsilon/2$ if the number of training data $N$ satisfies

$$
N > \overline{N}_{\min} \equiv 2MK_1/\varepsilon.
$$

(38)

Then both the expectation $\mathbb{E} \left[ \hat{e}_{L^2} \right]$ and the variance $\text{Var} \left[ \hat{e}_{L^2} \right]$ can be bounded by $\varepsilon$. Since $\varepsilon > 0$ is arbitrary, the proof is completed.

Recall that by applying arbitrary basis functions to represent $\hat{p}(y|\theta)$ in (5), it is typically not easy to evaluate the upper bound $D$ in (28) which implies that it is not easy to determine how many observation data, $N_{\min}$ [Eq. (29)], should be used for training. However, by applying the data-driven basis functions $\overline{\psi}_k$ to represent $\hat{p}(y|\theta)$ in (5), the minimum number of training data, $N_{\min}$ [Eq. (38)], becomes independent of $D$, and is only dependent of $M$, $K_1$, and $\varepsilon$ as can be seen from Theorem 2. To let the error induced by modes $k \leq K_1$ be smaller than a desired $\varepsilon/2$, we can easily determine how many observation data, $N_{\min}$ [Eq. (38)], should be used for training. In this sense, the specific data-driven Hilbert space $L^2(\mathbb{N}, \overline{\tau})$ with the corresponding basis functions $\overline{\psi}_k$ is a good choice for representing (5).

We have so far theoretically verified the validity of the representation (5) in estimating the conditional density $p(y|\theta_j)$ using any appropriate basis functions [Theorem 1]. In particular, using the data-driven basis
\( \psi_k(y) \in L^2(N, \eta) \), we can easily control the error of conditional density estimation by specifying the number of training data \( N \) [Theorem 2]. To summarize, the training procedures can be outlined as follows:

1-A Generate the training dataset including training parameters \( \{\theta_j\}_{j=1, \ldots, M} \) and observations \( \{y_{i,j}\}_{j=1, \ldots, M} \). The length of training data \( N \) is empirically determined based on the criteria (29) or (38).

1-B Construct the basis functions for parameter \( \theta \) space and for observation \( y \) space by using the training dataset. For \( y \) space, we need to empirically choose the number of basis functions \( K_1 \) to let the error induced by modes \( k > K_1 \) as small as possible. In particular, for the data-driven Hilbert space, we will provide a detailed discussion on how to estimate the data-driven basis functions of \( L^2(N, q) \) with the sampling density \( q \) from the training data in the following Sec. 4. Note that this basis estimation will introduce additional errors beyond the results in this section which assume the data-driven basis functions are given.

1-C Train the matrix \( C_{Y\Theta} \) in (7) and then estimate the conditional density \( \hat{p}(y|\theta) \) by using the nonparametric RKWHS representation (5) with the expansion coefficients \( \hat{c}_{Y\theta,k} \) (10).

1-D Finally, for new observations \( y^\dagger = \{y^\dagger_1, \ldots, y^\dagger_T\} \), define a likelihood function as a product of conditional densities of new observations \( y^\dagger \) given any \( \theta \),

\[
p(y^\dagger|\theta) \equiv \prod_{t=1}^{T} \hat{p}(y^\dagger_t|\theta).
\] (39)

Next, we address the second important question for the RKWHS representation [Procedure (1-B)]: how to construct basis functions for \( \theta \) and \( y \). Especially, we focus on how to construct data-driven basis functions for \( y \).

4 Basis functions

In this section, we address the main issue of how to construct basis functions for parameter \( \theta \) and observation \( y \). This section will be organized as follows. In Sec. 4.1, we discuss how to employ analytical basis functions for parameter \( \theta \) and for observation \( y \) as in the usual polynomial chaos expansion. In Sec. 4.2, we discuss how to construct the data-driven basis functions \( \psi_k \) of \( L^2(N, \eta) \) with \( N \) being the manifold of the training dataset \( \{y_{i,j}\}_{j=1, \ldots, M} \) and the weight \( \eta \) by (32) being the sampling density of \( \{y_{i,j}\}_{j=1, \ldots, M} \).

4.1 Analytic basis functions

First, we address the issue of how to choose the training parameters and how to construct the corresponding basis functions using these training parameters for \( \theta \) space. If no prior information about the parameter space is known, we can assume that the training parameters are uniformly distributed on the parameter \( \theta \) space. In particular, we choose \( M \) number of well-sampled training parameters \( \{\theta_j\}_{j=1, \ldots, M} = \{\theta_1^1, \ldots, \theta_m^1\}_{j=1, \ldots, M} \) in an \( m \)-dimensional box \( \mathcal{M} \subset \mathbb{R}^m \),

\[
\mathcal{M} \equiv [\theta_{\text{min}}^1, \theta_{\text{max}}^1] \times \cdots \times [\theta_{\text{min}}^m, \theta_{\text{max}}^m],
\] (40)

where \( \times \) denotes a Cartesian product, two parameters \( \theta_{\text{min}}^s \) and \( \theta_{\text{max}}^s \) are the minimum and maximum values of the uniform distribution for the \( s \)-th coordinate of \( \theta \) space. Here, the well-sampled uniform distribution corresponds to a regular grid, which is a tessellation of \( m \)-dimensional Euclidean space \( \mathbb{R}^m \) by congruent
Figure 1: (Color online) An example of well-sampled 2D uniformly distributed data points (blue circles). The boundary of the uniform distribution is depicted with a red square. Also, these well-sampled data points correspond to the training parameters in Example 1 in Sec. 5. In the Example, the well-sampled uniformly distributed training parameters are \((\sigma_{X_1}^2, \sigma_{X_2}^2) \in \{(i, j)\}_{i=5, \ldots, 12} \) (blue circles). The equal spacing distances of both coordinates are 1. The 2-dimensional box \(M\) is \([4.5, 12.5]^2\) (red square).

parallelotopes. Two parameters \(\theta_{s\min}^s\) and \(\theta_{s\max}^s\) are determined by

\[
\begin{align*}
\theta_{s\min}^s &= \min_{j=1, \ldots, M} \{\theta_j^s\} - \gamma \left( \max_{j=1, \ldots, M} \{\theta_j^s\} - \min_{j=1, \ldots, M} \{\theta_j^s\} \right), \\
\theta_{s\max}^s &= \max_{j=1, \ldots, M} \{\theta_j^s\} + \gamma \left( \max_{j=1, \ldots, M} \{\theta_j^s\} - \min_{j=1, \ldots, M} \{\theta_j^s\} \right),
\end{align*}
\]

For \(M\) regularly spaced grid points \(\theta_j^s\), we set \(\gamma = 0.5(M^s - 1)^{-1}\) in all of our numerical examples below, where \(M^s\) is the number of training parameters in the \(s\)th coordinate. For example, see Fig. 1 for the 2D well-sampled uniformly distributed data \(\{(5, 5), (6, 5), \ldots, (12, 12)\}\) (blue circles). In this case, the 2-dimensional box \(M\) is \([4.5, 12.5]^2\) (red square).

On this simple geometry, we will choose \(\varphi_k\) to be the tensor product of the basis functions on each coordinate. Notice that we have taken the weight function \(\tilde{q}\) to be the sampling density of the training parameters in order to simplify the expansion coefficient \(\hat{c}_{Y|\theta,k}\) in (10). In this case, the weight \(\tilde{q}\) is a uniform distribution on \(M\). Then for the \(s\)th coordinate of the parameter, \(\theta^s\), the weight function \(\tilde{q}^s(\theta^s)\) is a uniform distribution on the interval \([\theta^s_{\min}, \theta^s_{\max}]\) and one can choose the following cosine basis functions,

\[
\Phi_k(\theta^s) = \begin{cases} 
1, & \text{if } k^s = 0, \\
\sqrt{2} \cos \left( k^s \pi \frac{\theta^s - \theta_{s\min}^s}{\theta_{s\max}^s - \theta_{s\min}^s} \right), & \text{else,}
\end{cases}
\]

where \(\Phi_k(\theta^s)\) form a complete orthonormal basis of \(L^2([\theta^s_{\min}, \theta^s_{\max}], \tilde{q}^s)\). This choice of basis functions corresponds to exactly the data-driven basis functions produced by the diffusion maps algorithm on uniformly distributed data set on a compact interval, which will be discussed in Sec. 4.2. Although other choice such as the Legendre polynomials can be used, this choice will lead to a larger value of constant \(D\) in (29) that controls the minimum number of training data for accurate estimation.

Subsequently, we set \(L^2(M, \tilde{q}) = \bigotimes_{s=1}^m L^2([\theta_{s\min}^s, \theta_{s\max}^s], \tilde{q}^s)\), where \(\bigotimes\) denotes the Hilbert tensor product, \(\tilde{q}(\theta) = \bigotimes_{s=1}^m \tilde{q}^s(\theta^s)\) is the uniform distribution on the \(m\)-dimensional box \(M\). Correspondingly, the basis
functions $\varphi_k(\theta)$ are a tensor product of $\Phi_k(s^s)$ for $s = 1, \ldots, m$,

$$\varphi_k(\theta) = \bigotimes_{s=1}^{m} \Phi_k(s^s) = \Phi_k(\theta^1) \otimes \ldots \otimes \Phi_k(\theta^m),$$

(43)

where $k = (k^1, \ldots, k^m)$, and $\theta = (\theta^1, \ldots, \theta^m)$. Based on the property of the tensor product of Hilbert spaces, $\{\varphi_k(\theta)\}$ forms a complete orthonormal basis of $L^2(\mathcal{M}, \tilde{q})$.

We now turn to the discussion of how to construct analytic basis functions for $y$. The approach is similar to the one for parameter $\theta$, except that the domain of the data is specified empirically and the weight function is chosen to correspond to some well-known analytical basis functions, independent of the sampling distribution of the data $y$. That is, we assume the geometry of the data has the following tensor structure, $\mathcal{N} = \mathcal{N}^1 \times \ldots \times \mathcal{N}^n$, where $\mathcal{N}^s$ will be specified empirically based on the ambient space coordinate of $y$. Let $y^s$ be the $s$th ambient component of $y$, we can choose a weighted Hilbert space $L^2(\mathcal{N}^s, q^s(y^s; \alpha^s))$ with the weight $q^s$ depending on the parameters $\alpha^s$ and being normalized to satisfy $\int_{\mathcal{N}^s} q^s(y^s; \alpha^s)dy^s = 1$. For each coordinate, let $\Psi_{k^s}(y^s; \alpha^s)$ be the corresponding orthonormal basis functions which possess analytic expressions. Subsequently, we can obtain a set of complete orthonormal basis functions $\psi_k \in L^2(\mathcal{N}, q)$ for $y$ by taking the tensor product of these $\Psi_{k^s}$ as in (43).

For example, if the weight $q^s$ is uniform, $\mathcal{N}^s \subset \mathbb{R}$ is simply a one-dimensional interval. In this case, we can choose the cosine basis functions $\Psi_{k^s}$ for $y$ as in (42) such that the parameters $\alpha^s$ correspond to the boundaries of the domain $\mathcal{N}^s$, which can be estimated as in (41). In our numerical experiments below, we will set $\gamma = 0.1$. Another choice is to set the weight $q^s(y^s; \alpha^s)$ to be Gaussian. In this case, the domain is assumed to be the real line, $\mathcal{N}^s = \mathbb{R}$. For this choice, the corresponding orthonormal basis functions $\Psi_{k^s}$ are Hermite polynomials, and the parameters $\alpha^s$, corresponding to the mean and variance of the Gaussian distribution, can be empirically estimated from the training data.

In the remainder of this paper, we will always use the cosine basis functions for $\theta$. The application of (5) using cosine basis functions for $y$ is referred to as the Cosine representation. The application of (5) using Hermite basis functions for $y$ is referred to as the Hermite representation. In the next section, we introduce one method to construct data-driven basis functions $\overline{\psi}_k \in L^2(\mathcal{N}, \overline{q})$, which respects the geometry and the sampling distribution of the training data $\{y_{i,j}\}_{j=1,\ldots,M}$.

### 4.2 Data-driven basis functions

In this section, we discuss how to construct a set of data-driven basis functions $\overline{\psi}_k \in L^2(\mathcal{N}, \overline{q})$ with $\mathcal{N}$ being the manifold of the training dataset $\{y_{i,j}\}_{i=1,\ldots,N}$ and weight $\overline{q}$ in (32) being the sampling density of $\{y_{i,j}\}$ for all $i = 1, \ldots, N$, and $j = 1, \ldots, M$. The issues here are that the analytical expression of the sampling density $\overline{q}$ is unknown and the Riemannian metric inherited by the data manifold $\mathcal{N}$ from the ambient space $\mathbb{R}^n$ is also unknown. Fortunately, these issues can be overcome by the following diffusion maps algorithm [7, 4, 5].

#### 4.2.1 Learning the data-driven basis functions

Given a dataset $y_{i,j} \in \mathcal{N} \subseteq \mathbb{R}^n$ with the sampling density $\overline{q}(y)$ (32), defined with respect to the volume form inherited by the manifold $\mathcal{N}$ from the ambient space $\mathbb{R}^n$, one can use the kernel-based diffusion maps algorithm to construct an $MN \times MN$ matrix $L$ that approximates a weighted Laplacian operator, $L = \nabla \log (\overline{q}) \cdot \nabla + \Delta$, that takes functions with Neumann boundary conditions for compact manifold $\mathcal{N}$ with boundary if the manifold has boundary. The eigenvectors $\overline{\psi}_k$ of the matrix $L$ are discrete approximations of the eigenfunctions $\overline{\psi}_k(y)$ of the operator $L$, which form an orthonormal basis of the weighted Hilbert space $L^2(\mathcal{N}, \overline{q})$. Each component of the eigenvector $\overline{\psi}_k \in \mathbb{R}^{MN}$ is a discrete estimate of the eigenfunction $\overline{\psi}_k(y_{i,j})$, evaluated at the
training data point \( y_{i,j} \). The sampling density \( \bar{\eta} \) defined in (32) is estimated using a kernel density estimation method [3].

In contrast to the analytic continuous basis functions in above Sec. 4.1, the data-driven basis functions \( \bar{\psi}_k(y) \in L^2(\mathbb{N}, \bar{\eta}) \) are represented nonparametrically by the discrete eigenvectors \( \bar{\psi}_k \in \mathbb{R}^{MN} \) using the diffusion maps algorithm. The outcome for training is a discrete estimate of the conditional density, \( \hat{p}(y_{i,j} | \theta) \), which estimates the representation \( \hat{p}(y|\theta) \) (5) at each training data point \( y_{i,j} \).

In our implementation, we use the variable-bandwidth diffusion maps (VBDM) algorithm introduced in [4], which extends the diffusion maps to non-compact manifolds without boundary. See the supplementary material of [10] for the MATLAB code of this algorithm. We should point out that this discrete approximation induces errors in the basis function, which are estimated in detail in [4]. These errors are in addition to the error estimations in Sec. 3.

We note that if the data are uniformly distributed on a one-dimensional bounded interval, then VBDM solutions are the cosine basis functions, which are eigenfunctions of the Laplacian operator on bounded interval with Neumann boundary conditions. This means that the cosine functions in (42) that are used to represent each component of \( \theta \) are analog of the data-driven basis functions. The difference is that with parametric choice in (42), one avoids using VBDM in the expense of specifying the boundaries of the domain, \([\theta_{\min}^s, \theta_{\max}^s]\). In the remainder of this paper, we refer to an application of (5) with cosine basis functions for \( \theta \) and VBDM basis functions for \( y \) as the VBDM representation.

However, a direct application of the VBDM algorithm suffers from the expensive computation cost for large training data. In the next Sec. 4.2.2, we will address the issue of how to construct basis functions for large number of training data. Second issue arises from the discrete representation of the conditional density in the observation \( y \) space using the VBDM algorithm. Notice that the VBDM representation, \( \hat{p}(y_{i,j} | \theta) \), is only estimated at each training data point \( y_{i,j} \). A natural question is how to extend the representation onto new observations \( y_t \notin \{y_{i,j}\}_{j=1,...,M} \) that are not part of the training dataset [Procedures (1-C) and (1-D)]. In Sec. 4.2.3, we will introduce an extension method for approximating the representation on new observations.

### 4.2.2 Data reduction

When \( MN \) is very large, the VBDM algorithm becomes numerically expensive since it involves solving an eigenvalue problem of matrix size \( MN \times MN \). Notice that the number of training parameters \( M \) grows exponentially as a function of the dimension of parameter, \( m \), if well-sampled uniformly distributed training parameters are used. To overcome this large training data problem, we employ an empirical data reduction method to reduce the original \( MN \) training data points \( \{y_{i,j}\}_{i=1,...,N, j=1,...,M} \) to a small \( B (\ll MN) \) number of training data points yet preserving the sampling density \( \bar{\eta}(y) \) in (32). Subsequently, we apply the VBDM algorithm on these reduced training data points.

The basic idea of our method is that we first cluster the training dataset \( \{y_{i,j}\}_{i=1,...,N} \) into \( B \) number of boxes and then take the average of data points in each box as a reduced training data point. First, we cluster the training data \( \{y_{i,j}\} \), based on the ascending order of the 1\textsuperscript{st} coordinate of the training data, \( \{y_{1,j}^i\} \), into \( B^1 \) number of groups such that each group has the same number (= \( MN/B^1 \)) of data points. After the first clustering, we obtain \( B^1 \) groups with each group denoted by \( G^1_{k_1} \) for \( k_1 = 1,...,B^1 \). Here, the super-index 1 denotes the first clustering and the sub-index \( k_1 \) denotes the \( k_1 \)th group. Second, for each group \( G^1_{k_1} \), we cluster the training data \( \{y_{i,j}\} \) inside \( G^1_{k_1} \), based on the ascending order of the 2\textsuperscript{nd} coordinate of the training data, \( \{y_{1,j}^2\} \), into \( B^2 \) number of groups such that each group has the same number (= \( MN/B^1B^2 \)) of data points. After the second clustering, we obtain totally \( B^1B^2 \) groups with each group denoted by \( G^2_{k_1,k_2} \) for \( k_1 = 1,...,B^1 \), and \( k_2 = 1,...,B^2 \). We can operate such clustering \( n \) times, where \( n \) is the dimension of the observation \( y \) ambient space. After \( n \) times clustering, we obtain \( B \equiv \prod_{s=1}^{n} B^s \) groups with each group
denoted by $G^a_{n,k_1...k_n}$ with $k_s = 1,\ldots,B^s$, for all $s = 1,\ldots,n$. Each group is a box [see Fig. 2 for example]. After taking the average of the data points in each box $G^a_{n,k_1...k_n}$, we obtain $B$ number of reduced training data points. In the remainder of this paper, we denote these $B$ number of reduced training data points by $\{\bar{y}_b\}_{b=1,...,B}$ and refer to them as the box-averaged data points. Intuitively, this algorithm partitions the domain into hyperrectangle such that $P_t(\bar{y} \in G^a_{k_1...k_n}) \approx 1/B$. Note that the idea of our data reduction method is analogous to that of multivariate $k$-nearest neighbor density estimates [15, 17].

To examine whether the distribution of these box-averaged data points is close to the sampling density \( \bar{q}(y) \) of the original dataset, we apply our reduction method to several numerical examples in the following. Figure 2(a) shows the reduction result for few number (= 64) of uniform data in \([0, 1] \times [0, 1] \). Here, $B^1 = 4$ and $B^2 = 4$ so that there are total $B = 16$ boxes and inside each box there are 4 uniform data points (blue circles). It can be seen that the box-averaged data points (red circles) are far away from the well-sampled uniform data points (cyan crosses). However, when uniform data points (blue circles) increase to a large number (= 6400), box-average data points (red circles) are very close to well-sampled uniform data points (cyan crosses) as shown in Fig. 2(b). This suggests that these box-averaged data points nearly admit the uniform distribution when there are a large number of original uniform data points. Figs. 2(c) and (d) show the comparison of the kernel density estimates applied on the box-averaged data for different $B$ for standard normal distribution and the distribution proportional to $\exp\left[-\left(-X_1^2 + X_2^2 + X_3^2\right)\right]$, respectively. It can be seen that the reduced box-averaged data points nearly preserve the distribution of the original large dataset, $N = 64000, 000$.

When $MN$ is very large, the VBDM algorithm for the construction of data-driven basis functions [Procedure (1-B)] can be outlined as follows. We first use our data reduction method to obtain $B$ number of box-averaged data $\{\bar{y}_b\}_{b=1,...,B} \subseteq \mathbb{N} \subseteq \mathbb{R}^n$ with sampling density $\bar{q}(y) \approx \frac{1}{M} \sum_{j=1}^{M} p(y|\theta_j)$ in (32). The sampling density $\bar{q}$ is estimated at the box-averaged data $\bar{y}_b$ using all the box-averaged data points $\{\bar{y}_b\}_{b=1,...,B}$ by a kernel density estimation method. Implementing the VBDM algorithm, we can obtain orthonormal eigenvectors $\bar{\psi}_k \in \mathbb{R}^B$, which are discrete estimates of the eigenfunctions $\bar{\psi}_k(y) \in L^2(\mathbb{N}, \bar{q})$. The $b$th component of the eigenvector $\bar{\psi}_k$ is a discrete estimate of the eigenfunction $\bar{\psi}_k(\bar{y}_b)$, evaluated at the box-averaged data point $\bar{y}_b$. Due to the dramatic reduction of the training data, the computation of these eigenvectors $\bar{\psi}_k \in \mathbb{R}^B$ becomes much cheaper than the computation of the eigenvectors $\hat{\psi}_k \in \mathbb{R}^{MN}$ using the original training dataset $\{y_{i,j}\}_{i=1,...,N}$. Then we can obtain a discrete representation (5) of the conditional density at the box-averaged data points $\bar{y}_b$, $\hat{p}(\bar{y}_b|\theta)$.

### 4.2.3 Nyström extension

For the Procedure (1-C), when training the matrix $C_{Y\Theta}$ in (7), we need to know the estimate of the eigenfunction $\bar{\psi}_k(y_{i,j})$ at all the original training data $y_{i,j} \notin \{\bar{y}_b\}_{b=1,...,B}$. However, we only have the discrete estimate of the eigenfunction $\bar{\psi}_k(\bar{y}_b)$ at the box-averaged data points $\bar{y}_b$. This suggests that we need to extend the eigenfunctions $\bar{\psi}_k(\bar{y}_b)$ onto all the original training data $\{y_{i,j}\}_{j=1,...,M}$. Notice that this extension problem is the same as the aforementioned second issue for the VBDM algorithm: how to extend the representation onto new observations $y_t \notin \{y_{i,j}\}_{j=1,...,M}$ that are not part of the training dataset [Procedure (1-D)]. One solution to this issue is that we can first extend the eigenfunctions $\bar{\psi}_k(y_{i,j})$ to the new observation, $\bar{\psi}_k(y_t)$, and then represent the density, $\hat{p}(y_t|\theta)$, with a set of extended basis functions $\bar{\psi}_k(y_t)$.

For the convenience of discussion, the training data that are used to construct the eigenfunctions are denoted by $\{y_{r,old}\}_{r=1,...,R}$ and all the data that are not part of $\{y_{r,old}\}_{r=1,...,R}$ are denoted by $y_{new}$. To extend the eigenfunctions $\bar{\psi}_k(y_{r,old})$ onto the data point $y_{new} \notin \{y_{r,old}\}_{r=1,...,R}$, one approach would be to use the Nyström extension [22] that is based on the basic theory of RKHS [1]. Let $L^2(\mathbb{N}, \bar{q})$ be a RKWHS with a symmetric positive kernel $\tilde{T} : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{R}$. Then for any function $f \in L^2(\mathbb{N}, \bar{q})$, the Moore-Aronszajn
Figure 2: (Color online) Data reduction for (a) few number (= 64) of uniformly distributed data, and (b) many number (= 6400) of uniformly distributed data. The 64 blue circles correspond to uniformly distributed data, 16 cyan crosses correspond to well-sampled uniformly distributed data, and 16 red circles correspond to box-averaged data. Boxes are partitioned by horizontal and vertical black lines. The vertical black lines correspond to the first clustering and the horizontal lines correspond to the second clustering. Panels (c) and (d) display the comparison of kernel density estimates on the box-averaged data for different number $B$ for (c) standard normal distribution, and (d) the distribution proportional to $\exp[-(X_1^2 + X_2^2 + X_3^2)]$, respectively. For comparison, also plotted is the analytic probability density of the distribution. The total number of the points is 640000. It can be seen that the reduced box-averaged data points nearly preserve the distribution of the original dataset.
Consider an Ornstein–Uhlenbeck process as follows:

\[ \frac{dX}{dt} = -\frac{1}{2}X + \frac{1}{2} \Sigma^{1/2}dW_t, \]  

(45)

where the non-symmetric kernel function \( T : \mathcal{N} \times \mathcal{N} \rightarrow \mathbb{R} \) is related to the symmetric kernel \( \tilde{T} \) by \( T(y_i, y_j) = \tilde{T}(y_i, y_j)\tilde{\eta}^{1/2}(y_j) \) with \( \tilde{\eta}(y_i) \) being the sampling density of \( \{y_r^{\text{old}}\}_{r=1}^R \) at \( y_i \). See the detailed evaluation of the kernels \( \tilde{T} \) and \( T \) for the Nyström extension in [11]. After obtaining the estimate of the eigenfunction \( \tilde{\psi}_k(y^{\text{new}}) \) using the Nyström extension, we can train the matrix \( C_{Y\Theta} \) in (7) for large \( MN \) and then obtain the representation of the conditional density on arbitrary new observation \( y_i, \tilde{p}(y_i|\theta) \).

To summarize this section, we have constructed two different sets of basis functions for \( \mathbf{y} \), the analytic basis functions of \( L^2(\mathcal{N}, q) \) such as the Hermite and Cosine basis functions which assume that the manifold is \( \mathbb{R}^n \) or hyperrectangle, respectively, and the data-driven basis functions of \( L^2(\mathcal{N}, \tilde{\eta}) \) with \( \mathcal{N} \) being the data manifold and \( \tilde{\eta} \) being the sampling density that are computed using VBDM algorithm. In the next section, we will compare the numerical performances of the Hermite, Cosine, and VBDM basis functions for parameter estimation application.

## 5 Parameter estimation using the Metropolis scheme

Here, we briefly review the Metropolis scheme for estimating the posterior density \( p(\theta|y^\dagger) \) given new observations \( y^\dagger = \{y_1^\dagger, \ldots, y_T^\dagger\} \) for a specific parameter \( \theta^\dagger \). The key idea of the Metropolis scheme is to construct a Markov chain such that it converges to samples of conditional density \( p(\theta|y^\dagger) \) as the target density. In our application, the parameter estimation procedures can be outlined as follows:

\begin{enumerate}
  \item (2-A) Suppose we have \( \theta_0 \sim p(\theta_0|y^\dagger) > 0 \), then for \( i \geq 1 \) we can sample \( \theta^\ast \sim \kappa(\theta_{i-1}, \theta^\ast) \). Here, \( \kappa \) is the proposal kernel density. For example, use the random walk Metropolis algorithm to generate proposals, \( \kappa(\theta_{i-1}, \theta^\ast) = \mathcal{N}(\theta_{i-1}, C) \), where \( C \), the proposal covariance, is a tunable nuisance parameter.
  \item (2-B) Accept the proposal, \( \theta_i = \theta^\ast \) with probability \( \min(\frac{p(\theta^\ast|y^\dagger)}{p(\theta_{i-1}|y^\dagger)}, 1) \), otherwise set \( \theta_i = \theta_{i-1} \). Repeat Procedures (2-A) and (2-B) above. Notice that the posterior \( p(\theta|y^\dagger) \) can be determined from the prior \( p_0(\theta) \) and the likelihood \( p(y^\dagger|\theta) \) based on the Bayes’ theorem (3). The likelihood function \( p(y^\dagger|\theta) \) is defined as a product of conditional densities of new observations \( y^\dagger = \{y_1^\dagger, \ldots, y_T^\dagger\} \) in (39) [Procedure (1-D)]. The conditional densities of new observations \( y^\dagger \) given \( \theta \) are obtained from the training Procedure (1-C).
  \item (2-C) Generate a sufficiently long chain and use chain’s statistic as an estimator of the true parameter \( \theta^\dagger \). Take multiple runs of the chain started at different initial \( \theta_0 \), and examine whether all these runs converge to the same distribution. The convergence of all the examples below have been validated using 10 randomly chosen different initial conditions.
\end{enumerate}

### 5.1 Example I: Two-dimensional Ornstein–Uhlenbeck (OU) process

Consider an Ornstein–Uhlenbeck process as follows:

\[ dX = -\frac{1}{2}Xdt + \Sigma^{1/2}dW_t, \]  

(45)

where \( X \equiv (X_1, X_2)^\top \) denotes the state variable, \( W_t = (W_1, W_2)^\top \) denotes two-dimensional Wiener processes, and \( \Sigma \in \mathbb{R}^{2\times 2} \) is a diagonal matrix with main diagonal components \( \sigma_{X_1}^2 \) and \( \sigma_{X_2}^2 \) to be estimated. In
stationary, the solution of Eq. (45) \( X = (X_1, X_2)^\top \) admits a Gaussian distribution \( X \sim \mathcal{N}(0, \Sigma) \),

\[
p(X|\Sigma) = \det(2\pi \Sigma)^{-\frac{1}{2}} \exp \left[ -\frac{1}{2} X^\top \Sigma^{-1} X \right].
\] (46)

Our goal here is to estimate the posterior density and the posterior mean of the parameters \((\sigma_{X_1}^2, \sigma_{X_2}^2)\), given a finite number, \( T \), of observations, \( X^\dagger \equiv (X_1^{\dagger}, \ldots, X_T^{\dagger}) \), for hidden true parameters \((\sigma_{X_1}^2, \sigma_{X_2}^2)^\dagger\) = (6.5, 6.3).

One can show that the likelihood function for this problem is the inverse matrix gamma distribution, \( \Sigma \sim \text{IMG} \left( \frac{T}{2} - \frac{3}{2}, 2, \Psi \right) \), where \( \Psi = X^\dagger (X^\dagger)^\top \in \mathbb{R}^{2 \times 2} \). If a prior is defined to be also the inverse matrix gamma distribution, \( \Sigma \sim \text{IMG} (\alpha_0, 2, 0) \), for some value of \( \alpha_0 \), then the posterior density \( p(\Sigma|X^\dagger) \) can be obtained by applying the Bayes’ theorem,

\[
p(\Sigma|X^\dagger) \sim \text{IMG} \left( \alpha_0 + \frac{T}{2}, 2, \Psi \right).
\] (47)

The posterior mean can thereafter be obtained as,

\[
(S)^{PM} = \begin{pmatrix}
(\sigma_{X_1}^2)^{PM} \\
0
\end{pmatrix}, \quad \begin{pmatrix}
0 \\
(\sigma_{X_2}^2)^{PM}
\end{pmatrix} = \frac{\Psi}{T + 2\alpha_0 - 3}.
\] (48)

See Section 7 in the supplementary material for the detail derivation of the posterior density function and the posterior mean.

To compare with the analytic conditional density \( p(X|\Sigma) \) (46), we learn three RKWHs representations of the conditional density function, \( \hat{p}(X|\Sigma) \), by using the same training dataset. For training, we use \( M = 64 \) well-sampled uniformly distributed training parameters [shown in Fig. 1], \((\sigma_{X_1}^2, \sigma_{X_2}^2)\), where \( \sigma_{X_j}^2 \in \{5, 6, \ldots, 12\} \), which are denoted by \( \Sigma_j \). For each training parameter \( \Sigma_j \), we generate \( N = 640000 \) number of well-sampled normally distributed observation data. For Hermite and Cosine representations, we use 20 number of basis functions for each coordinate, and then we can construct \( K_1 = 400 \) basis functions of two-dimensional observation, \( X \), by taking the tensor product. For the VBDM representation, we first compute \( B = B_1 \times B_2 = 100 \times 100 \) box-averaged data by our data reduction method [Sec. 4.2.2]. After reducing the number of training data from \( MN \) to \( B \), we can learn \( K_1 = 400 \) data-driven basis functions from \( B \) box-averaged data using the VBDM algorithm [4].

Figure 3(a) displays the analytic conditional density (46) and Figs. 3(b)-(d) display the pointwise errors of the conditional densities \( \hat{p}(X|\Sigma) \equiv p(X|\Sigma) - \hat{p}(X|\Sigma) \) at the training parameter \((\sigma_{X_1}^2, \sigma_{X_2}^2) = (5, 5)\). It can be seen from Figs. 3(b)-(d) that all the pointwise errors are small compared to the analytic \( p(X|\Sigma) \) in Fig. 3(a) so that all representations of conditional densities \( \hat{p}(X|\Sigma) \) are in excellent agreement with the analytic
p(X|Σ) [Fig. 3(a)]. This suggests that for the Hermite representation, the upper bound $D$ (28) in Theorem 1 is finite so that the representation is valid in estimating the conditional density as can be seen from Fig. 3(b). On the other hand, the upper bound $D$ for the Cosine and the VBDM representations are always finite as mentioned in Remark 2 and Proposition 2, respectively. We should also point out that for this example, the VBDM representation performs the worst with errors of order $10^{-4}$ compared to the Hermite and Cosine representations whose errors are on the order of $10^{-6}$. This larger error in the VBDM representation is because the data-driven basis functions are estimated by discrete eigenvectors $\psi_k \in \mathbb{R}^B$, so additional errors [4] are introduced through this discrete approximation (especially on the high modes) on the box-averaged data, $\{Y_b\}_{b=1,...,B}$. $B = 10000$. On the other hand, for Hermite and Cosine representations, their analytic basis functions are known so that the errors can be approximated by (20) in Theorem 1. See Section 7 in the supplementary material for more discussions about the numerical results for this example.

We now estimate the posterior density (47) and mean (48) by using MCMC method [Procedures (2-A) - (2-C)]. We generate $T = 400$ well-sampled normally distributed data as the observations from the true values of variance $\Sigma^| = (\sigma^2_{X_1}, \sigma^2_{X_2}) = (6.5, 6.3)$. From the analytical formula (48), we obtain the posterior mean as $(\sigma^2_{X_1}, \sigma^2_{X_2})_{PM} = (6.03, 5.84)$. Here, the posterior mean deviates greatly from the true value since we only use $T = 400$ normally distributed observation data as new observations. If using much more new observation data, the analytical posterior mean (48) will get closer to the true value, $(\sigma^2_{X_1}, \sigma^2_{X_2})$. In our simulation, we set the parameter in the prior, $\alpha_0 = 1$, and the proposal covariance, $C = 0.01I$. For each chain, the initial condition $(\sigma^2_{X_1,0}, \sigma^2_{X_2,0})$ is drawn randomly from $U[5,12]^2$ and 800000 iterations are generated for the chain.

Figures 4(b), (c), and (d) display the densities of the chain by using Hermite, Cosine, and VBDM representation, respectively. The densities are plotted using kernel density estimate on the chain ignoring the first 10000 iterations. For comparison, Fig. 4(a) displays the analytic posterior density (47). It can be seen from Fig. 4 that the posterior densities by three representations are in excellent agreement with each other and with the analytic posterior density (47). Figure 4 also shows the comparison between the posterior mean (48) and the MCMC mean estimates. From our numerical results, MCMC mean estimates by all representations and the analytic posterior mean (48) are identical within numerical accuracy. Therefore, for this 2D OU-process example, all representations are valid in estimating the posterior density and posterior mean of parameter $\Sigma$.

Next, we will investigate a system for which the intrinsic dimension $d$ is less than the dimension of ambient space $n$. From the system, we will find that representations exhibit quite different performances between analytic and data-driven basis functions in estimating the posterior density.
5.2 Example II: Three-dimensional system of SDE’s on a torus

Consider a system of SDE’s on a torus defined in the intrinsic coordinates \((\theta, \phi) \in [0, 2\pi)^2\)

\[
d\begin{pmatrix} \theta \\ \phi \end{pmatrix} = a(\theta, \phi) \, dt + b(\theta, \phi) \begin{pmatrix} dW_1 \\ dW_2 \end{pmatrix},
\]

(49)

where \(W_1\) and \(W_2\) are two independent Wiener processes, and the drift and diffusion coefficients are

\[
a(\theta, \phi) = \begin{pmatrix} \frac{1}{4} + \frac{1}{3} \cos(\theta) \cos(2\phi) + \frac{1}{3} \cos(\theta + \pi/2) \\ 10 + \frac{1}{2} \cos(\theta + \phi/2) + \cos(\theta + \pi/2) \end{pmatrix},
\]

\[
b(\theta, \phi) = \begin{pmatrix} \frac{1}{3} \cos(\theta + \phi) \\ \frac{1}{3} \cos(\theta + \phi/2) + \cos(\theta + \pi/2) \end{pmatrix}.
\]

The initial condition is \((\theta, \phi) = (\pi, \pi)\). Here, \(D\) is a parameter to be estimated. This example exhibits non-gradient drift, anisotropic diffusion, and multiple time scales. The training dataset are generated by numerically solving the SDE in (49) with a time step \(\Delta t = 0.1\) and then mapping this data into the ambient space, \(\mathbb{R}^3\), via the standard embedding of the torus given by

\[
x \equiv (x, y, z) = ((2 + \sin(\theta)) \cos(\phi), (2 + \sin(\theta)) \sin(\phi), \cos(\theta)).
\]

(50)

Here, \(x \equiv (x, y, z)\) are observations. This system on a torus satisfies \(d < n\), where \(d = 2\) is the intrinsic dimension of \(x\) and \(n = 3\) is the dimension of ambient space \(\mathbb{R}^n\). Our goal is to estimate the posterior density and the posterior mean of parameter \(D\) given discrete-time observations of \(x^\dagger\) for a specific parameter \(D^\dagger\).

For training, we use \(M = 8\) well-sampled uniformly distributed training parameters, \(\{D_j = j/4\}_{j=1}^8\). For each training parameter \(D_j\), we generate \(N = 54000\) observations of \(x\). For Hermite and Cosine representation, we construct 10 basis functions for each \(x, y, z\) coordinate in Euclidean space. After taking tensor product of these basis functions, we can obtain \(K_1 = 1000\) basis functions on the ambient space \(\mathbb{R}^3\). For VBDM representation, we first compute \(B = B_1 \times B_2 \times B_3 = 30^3\) box-averaged data points by our data reduction method. However, we find that some of the \(B\) box-averaged data points are far away from the torus. After ignoring these points, we eventually choose \(B = 26020\) of the box-averaged data points that are close enough to the torus for training. Then, we learn \(K_1 = 1000\) data-driven basis functions on \(N\) from these 26020 box-averaged data points using the VBDM algorithm. For diagnostic comparisons, we construct another representation \(\tilde{p}(x|D_j)\), named intrinsic Fourier representation, which can be regarded as an accurate approximation of \(p(x|D_j)\), as it uses basis functions defined on the intrinsic coordinates \((\theta, \phi)\) instead of \(x \in \mathbb{R}^3\). See Section 8 in detail for the construction and the convergence of the intrinsic Fourier representation.

Figure 5 displays the comparison of the density estimates. It can be observed from Fig. 5 that the VBDM representation is in good agreement with the intrinsic Fourier representation, whereas Hermite and Cosine representations of \(\tilde{p}(x|D_j)\) deviate significantly from the intrinsic Fourier representation. The reason in short is that if the density \(p(\theta, \phi|D)\) in \((\theta, \phi)\) coordinate is in \(L^2([0, 2\pi)^2)\), then the corresponding VBDM representation with respect to \(dV(x)\) is in \(L^2(N, \mathbf{1})\). However, the representation [Hermite and Cosine] with respect to \(d\mathbf{x}\), \(x \in \mathbb{R}^3\) is not in \(L^2(\mathbb{R}^3, q^{-1})\). A more detailed explanation of this assertion is presented in Section 8.

We now compare the MCMC estimates with the true value, \(D^\dagger = 0.9\) from \(T = 10000\) observations. For this simulation, we set the prior to be uniformly distributed and empirically choose \(C = 0.01\) for the proposal. Figure 6 displays the posterior densities of the chains for all representations (each density estimate is smoothed with KDE on a chain of length 400000). Also displayed is the comparison between the true value \(D^\dagger\) and the MCMC mean estimates by all representations. Here, the mean estimate by the intrinsic Fourier representation nearly overlaps with the true value \(D^\dagger = 0.9\) as shown in Fig. 6. The mean estimate by the VBDM representation is closer to the true value \(D^\dagger\) compared to the estimates by Hermite and Cosine.
Figure 5: (Color online) Comparison of the conditional densities \( \hat{p}(x|D_j) \) estimated by using Hermite representation (first row), Cosine representation (second row), VBDM representation (third row), and intrinsic Fourier representation (fourth row). The left (a)(d)(g)(j), middle (b)(e)(h)(k), and right (c)(f)(i)(l) columns correspond to the densities on the training parameters \( D_1 = 0.25, D_4 = 1.00, \) and \( D_7 = 1.75 \), respectively. \( K_1 = 1000 \) basis functions are used for all representations. For fair visual comparison, all conditional densities are plotted on the same box-averaged data points, and normalized to satisfy \( \frac{1}{B} \sum_{b=1}^{B} \hat{p}(x_b|D_j)/\tilde{q}(x_b) = 1 \) with \( \tilde{q} \) being the estimated sampling density of the box-averaged data \( \{x_b\}_{b=1}^{B} \).
representations. Moreover, it can be seen from Fig. 6 that the posterior by the VBDM representation is close to the posterior by intrinsic Fourier representation, whereas posterior densities by Hermite and Cosine representation are not. This result suggests that when the intrinsic dimension is less than the ambient space dimension, $d < n$, the VBDM representation with data-driven basis functions in $L^2(N, \pi)$ is superior compared to the representations with analytic basis functions.

## 5.3 Example III: Five-dimensional Lorenz-96 model

Consider a five-dimensional Lorenz-96 model [16]

$$\frac{dx_j}{dt} = x_{j-1}(x_{j+1} - x_{j-2}) - x_j + F, \quad j = 1, \ldots, 5,$$

(51)

with periodic boundary, $x_{j+5} = x_j$. The initial condition has a standard normal distribution for each $x_j$ and is kept the same for all training parameters. Our goal here is to estimate the posterior density and posterior mean of the hidden parameter $F$ given a time series of noisy observations $u = (u_1, u_2, u_3, u_4, u_5)$, $u_j(t_m) = x_j(t_m) + \epsilon_{m,j}$, $\epsilon_{m,j} \sim N(0, \sigma_j^2)$, $m = 1, \ldots, T$, where the noise variance $\sigma_j^2 = 2.5e^{-3}\text{Var}[x_j]$. Here, $x_j(t_m)$ denotes the approximate solution (with Runge-Kutta method) at discrete times $t_m = 20m\Delta t$, where $\Delta t = 0.05$ is the integration time step.

For training, we use $M = 8$ well-sampled uniformly distributed training parameters, $\{F_j = 7.55 + 0.1j\}_{j=1}^8$. For each training parameter $F_j$, we generate $N = 500$ noisy observations $u$. For Hermite and Cosine representation, we construct 5 Hermite and cosine basis functions for each coordinate, which yields a total of $K_1 = 5^5 = 3125$ basis functions in $\mathbb{R}^5$. For the VBDM representation, we directly apply the VBDM algorithm to learn $K_1 = 3125$ data-driven basis functions on manifold $N$ from $MN = 4000$ training dataset. From the VBDM algorithm, the estimated intrinsic dimension is $d = 2.19$ which is smaller than the dimension of the ambient space $n = 5$.

Figure 7 displays the comparison of the conditional densities $\hat{p}(u|F_j)$ using Hermite (first row), Cosine (second row), and VBDM (third row) representations. For this example, the Hermite and Cosine representations of conditional densities $\hat{p}(u|F_j)$ have smaller density values, compared to that of the VBDM representation.
Figure 7: (Color online) Comparison of the conditional densities \( \hat{p}(u|F_j) \) estimated by using Hermite representation (first row), Cosine representation (second row), and VBDM representation (third row). The left (a)(d)(g), middle (b)(e)(h), and right (c)(f)(i) columns correspond to the densities on the training parameters \( F_1 = 7.65, F_4 = 7.95, \) and \( F_7 = 8.25 \), respectively. \( K_1 = 3125 \) basis functions are used for all representations. For fair visual comparison, all conditional densities are plotted on the training dataset \( \{u_{i,j}\}_{i=1,...,N}^{j=1,...,M} \) on the coordinates \( (u_1, u_3, u_5) \), and normalized to satisfy \( \frac{1}{MN} \sum_{j=1}^{M} \sum_{i=1}^{N} \hat{p}(u_{i,j}|F_k)/\bar{q}(u_{i,j}) = 1 \) with \( \bar{q} \) being the estimated sampling density of the training dataset. For each \( \hat{p}(u|F_j) \), only the largest 500 conditional densities are depicted with marker size 20 and the rest 3500 small conditional densities are depicted with marker size 1.
For the VBDM representation, the conditional densities $\tilde{p}(u|F_j)$ have higher peaks as depicted by the magenta circles [see Figs. 7(g)-(i)]. This indicates that the manifolds of training data $\{u_{i,j}\}_{i=1,...,N_j}$ are compact. By comparing Figs. 7(g), (h), and (i), it can be further observed that the magenta points, corresponding to the several largest conditional densities, are different for each $\tilde{p}(u|F_j)$. This indicates that the manifolds $N_j$ change and nearly have no intersection for different training parameter $F_j$. This is the reason why in this example we do not need to use box-averaged data since one can distinguish the training parameter $F_j$ only from small size of data by comparing the data manifolds $N_j$. This is the opposite of the previous Example II, where the data manifolds for all training parameters are exactly the same so that more training data are typically needed to distinguish the training parameters by comparing the conditional densities.

We now compare the MCMC mean estimates of the true value, $F^* = 8$ from $T = 1000$ noisy observations $u(t_m)$. Similar to the previous example, we also use a uniform prior distribution and $C = 0.01$ for the proposal. We generate the chain for 400000 iterations. Figure 8 displays the posterior densities of the chains for all representations. Also displayed is the comparison between the true value $F^*$ and the MCMC mean estimates by all representations. It can be seen from Fig. 8 that the mean estimate by VBDM representation is very close to the true value $F^*$, compared to the mean estimates by Hermite and Cosine representations. Notice also that although the mean estimate from the Cosine representation is not too far from the true parameter value, its posterior density is bimodal, which suggests more uncertainties on the mean estimates. Based on this numerical result, where the estimated intrinsic dimension $d \approx 2$ of the observations is lower than the ambient space dimension $n = 5$, again, the data-driven VBDM representation is superior compared to the Hermite and Cosine representations.

6 Conclusion

We have developed a framework of parameter estimation approach where MCMC is employed with a non-parametric likelihood function. Our approach approximates the likelihood function using the kernel embedding of conditional distribution formulation based on RKWHS. By analyzing the error estimation in Theorem 1, we have verified the validity of our RKWHS representation of the conditional density as long as $p(y|\theta) \in L^2(\mathcal{N}, q^{-1})$ and $\text{Var}_{\theta} [\psi_k(Y)]$ is finite. Furthermore, the analysis suggests that if the weight $q$ is chosen to be the sampling density of the data, the $\text{Var}_{\theta} [\psi_k(Y)]$ is always finite. This justifies the use of Variable Bandwidth Diffusion Maps (VBDM) for estimating the data-driven basis functions of the Hilbert space weighted by the sampling density on the data manifold.
To overcome the difficulty induced by large training data in the VBDM algorithm, we have provided a data reduction method that keeps the sampling distribution of the original data. We have compared the Hermite, Cosine, and VBDM representations in parameter estimation application on three instructive numerical examples. In the first example, where the dimension of the data manifold is exactly the dimension of the ambient space, \( d = n \), all three representations are valid in estimating the posterior density of parameter. However, in the examples where the dimension of the data manifold is strictly less than the dimension of the ambient space, \( d < n \), only VBDM representation can provide accurate estimation of the true parameter value.

These theoretical and numerical results inspire us to apply VBDM representation in many other parameter estimation applications. The next challenge will be to overcome a very high-dimensional problem, where one needs to identify the relevant features for estimating the parameters and construct the conditional densities on the relevant coordinates instead of on the full high-dimensional data space. This is the subject of our future work.

7 Appendix for Example I

This section provides more detailed discussions on the derivation of the explicit solutions of Example I and discusses additional numerical results.

In stationary, the solution of Eq. (45) \( \mathbf{X} = (X_1, X_2)^\top \) admits a Gaussian distribution \( \mathbf{X} \sim \mathcal{N}(\mathbf{0}, \Sigma) \) with the variance

\[
\Sigma = \begin{pmatrix}
\sigma_{X_1}^2 & 0 \\
0 & \sigma_{X_2}^2
\end{pmatrix},
\tag{52}
\]

so that the corresponding conditional density function \( p(\mathbf{X}|\Sigma) \) is

\[
p(\mathbf{X}|\Sigma) = \det(2\pi\Sigma)^{-\frac{1}{2}} \exp \left[ -\frac{1}{2} \mathbf{X}^\top \Sigma^{-1} \mathbf{X} \right].
\tag{53}
\]

To estimate the posterior density and the posterior mean of the parameters \( (\sigma_{X_1}^2, \sigma_{X_2}^2) \) in (52), we are given a time series of \( T \) observations, \( \mathbf{X}^\dagger \equiv (\mathbf{X}^\dagger_1, \ldots, \mathbf{X}^\dagger_T) \), for a specific variance \( \Sigma^\dagger \). Here, each observation \( \mathbf{X}^\dagger_i \) has two components \( \mathbf{X}^\dagger_i = \left(X^\dagger_i, X^\dagger_2\right)^\top \) corresponding to the \( i \)th observation of \( (X_1, X_2)^\top \).

Next, we derive the analytic posterior density and the posterior mean of the parameters \( (\sigma_{X_1}^2, \sigma_{X_2}^2) \) in (52). The likelihood function of the parameter \( \Sigma \) given \( \mathbf{X}^\dagger \) is,

\[
p(\mathbf{X}^\dagger|\Sigma) = \prod_{i=1}^{T} p(\mathbf{X}^\dagger_i|\Sigma) = \det(2\pi\Sigma)^{-\frac{T}{2}} \exp \left[ -\frac{1}{2} \text{tr}(\Psi\Sigma^{-1}) \right],
\tag{54}
\]

where \( \Psi \) is a \( 2 \times 2 \) matrix,

\[
\Psi = \mathbf{X}^\dagger(\mathbf{X}^\dagger)^\top = \begin{pmatrix}
\sum_{i=1}^{T} (X^\dagger_1)^2 & 0 \\
0 & \sum_{i=1}^{T} (X^\dagger_2)^2
\end{pmatrix}.
\tag{55}
\]

Note that the likelihood function (54) is an inverse matrix gamma distribution on \( \Sigma \). Recall that the density function of the three-parameter inverse matrix gamma distribution \( \Sigma \sim IMG(\alpha, \beta, \Psi) \) is,

\[
p(\Sigma) \propto |\Sigma|^{-(\alpha+3)/2} \exp \left[ -\frac{1}{\beta} \text{tr}(\Psi\Sigma^{-1}) \right].
\]
In particular, the likelihood function in (54) corresponds to the following inverse matrix gamma distribution

$$\Sigma \sim IMG \left(\frac{T}{2} - \frac{3}{2}, 2, \Psi \right).$$  \hspace{1cm} (56)

If a prior is defined to be also the inverse matrix gamma distribution, $$\Sigma \sim IMG (\alpha_0, 2, 0),$$

$$p_0(\Sigma) \propto |\Sigma|^{-\alpha_0-3/2},$$  \hspace{1cm} (57)

for some value of $$\alpha_0$$, then the posterior density can be obtained by applying the Bayes’ theorem,

$$p(\Sigma | X) \propto p_0(\Sigma) p(X | \Sigma) \sim IMG \left(\alpha_0 + \frac{T}{2}, 2, \Psi \right).$$  \hspace{1cm} (58)

After the normalization of (58), the posterior density of $$(\sigma_{X_1}^2, \sigma_{X_2}^2)$$ can be obtained as

$$p(\Sigma | X) = p((\sigma_{X_1}^2, \sigma_{X_2}^2) | (X_1, \ldots, X_T))$$

$$= \frac{(\Psi_1)^{\alpha+1}(\Psi_2)^{\alpha+1}}{2^{2\alpha+1}} \exp \left\{ -\frac{1}{2} \left( \frac{\Psi_1}{\sigma_{X_1}^2} + \frac{\Psi_2}{\sigma_{X_2}^2} \right) \right\},$$  \hspace{1cm} (59)

where

$$\alpha = \alpha_0 + \frac{T}{2},$$

and $$\Psi_1$$ and $$\Psi_2$$ are the diagonal components of the matrix $$\Psi$$ in (55),

$$\Psi_1 = \sum_{i=1}^{T} (X_i)^2, \quad \Psi_2 = \sum_{i=1}^{T} (X_i)^2.$$

The posterior mean can thereafter be obtained as,

$$(\sigma_{X_1}^2)_{PM} = \frac{\Psi_1}{2\alpha - 3} = \frac{\Psi_1}{T + 2\alpha_0 - 3}, \quad (\sigma_{X_2}^2)_{PM} = \frac{\Psi_2}{2\alpha - 3} = \frac{\Psi_2}{T + 2\alpha_0 - 3}. \hspace{1cm} (60)$$

To quantify the accuracy of $$\hat{p}(X | \Sigma)$$ for three representations, plotted in Figs. 3(b), (c), and (d) are the pointwise errors of the conditional densities $$\hat{e}(X | \Sigma) \equiv p(X | \Sigma) - \hat{p}(X | \Sigma)$$ at $$(\sigma_{X_1}^2, \sigma_{X_2}^2) = (5, 5)$$ for Hermite, Cosine, and VBDM representations, respectively. It can be seen from Figs. 3(b) and (c) that the errors of Hermite and Cosine representations are relatively small and are on order of $$10^{-6}$$. However, the error of VBDM representation is relatively large and is on order of $$10^{-4}$$ [Fig. 3(d)]. To further compare these representations, we investigate how the $$l^2$$-norm error $$\hat{e}_{l^2}$$ depends on the number of basis functions $$K_1$$. Here, the $$l^2$$-norm error $$\hat{e}_{l^2}$$ is defined by taking the summation of errors from all the training parameters,

$$\hat{e}_{l^2} = \left( \frac{1}{B \sum_{b=1}^{B} |X_b|} \left| \hat{e}(X_b | \Sigma) \right|^2 \right)^{1/2},$$  \hspace{1cm} (61)

where $$X_b$$ are the box-averaged data, $$\{\Sigma_j\}_{j=1}^{M}$$ are training parameters, $$\overline{\Psi}$$ is the sampling density of $$\{X_b\}_{b=1}^{B}$$ estimated using a kernel density estimation [3]. For $$\hat{e}(X | \Sigma)$$, the RKWH representation $$\hat{p}(X | \Sigma)$$ in (5) is computed by using different number of modes, $$K_1$$. Note that this definition of $$\hat{e}_{l^2}$$, similar to Frobenius norm, is a Monte Carlo approximation of the $$L^2$$-norm error $$\hat{e}_{l^2}$$ [Eq. (19)]. Figure 9 displays the $$l^2$$-norm error $$\hat{e}_{l^2}$$ as a function of $$K_1$$. It can be seen from Fig. 9 that when many basis functions are used ($$K_1 \approx 200$$), for Hermite and Cosine representations, their $$l^2$$-norm errors $$\hat{e}_{l^2}$$ are very small and are 0.04 and 0.02, respectively, whereas for VBDM representation, $$\hat{e}_{l^2}$$ is relatively large and is 0.7. The reason is that for the VBDM representation, the data-driven basis functions are estimated by the discrete eigenvectors $$\overline{\Psi}_k \in \mathbb{R}^B$$ so additional errors [4]
are induced in this function approximation that uses only finite number of box-averaged data, \( \{ X_b \}_{b=1}^{B}, \) with \( B = 10000, \) especially for eigenfunctions that are more oscillatory. To clarify this, we should point out that the pointwise error estimate for the VBDM representation at each \( X_b \) is bounded above by the ratio \( \| \nabla \psi_k (X_b) \| \sqrt{B} \) (see [4] for details). This implies that for more oscillatory functions \( \psi_k \) for large \( k \) (e.g., Fourier series), larger \( B \) is required to control the magnitude of the gradient in this error bound. In contrast, for Hermite and Cosine representations, their analytic basis functions are known so that the errors \( \hat{e}_{l^2} \) can be approximated by (20) in Theorem 1. However, when only few basis functions are used \( (K_1 \approx 10) \), it can be seen from Fig. 9 that the error \( \hat{e}_{l^2} \) by VBDM representation and Hermite representation is relatively small compared to the error by Cosine representation. This is one superiority of VBDM representation, that is, VBDM representation can well approximate the density using only few leading eigenfunctions since these VBDM eigenfunctions are data-driven. For Hermite representation, its weight function happens to be very close to the sampling density \( q \) for this OU example, so that Hermite representation can well approximate the density with only few Hermite polynomials. For Cosine representation, since its weight function deviates greatly from the sampling density \( q \), it can be expected that Cosine representation can only provide a fair approximation with few cosine basis functions. As can be seen in Example III in the paper, this observation plays an important role when the data manifold is a sub-manifold of \( \mathbb{R}^n \) with intrinsic dimension much less than \( n \), since the computations with analytic representation becomes very expensive even for \( n \approx 5 \). However, for VBDM representation, only very few basis functions are needed to represent the density.

It is worth mentioning that for VBDM representation, although the \( l^2 \)-norm error \( \hat{e}_{l^2} \) is relatively large \( (\sim 0.7) \) [Fig. 9] and can not converge to zero due to the additional error induced by the finite dimension of eigenvectors, the eventual posterior density and posterior mean can still be well approximated as shown in Fig. 4. This suggests that in order to well approximate the posterior, the error is only required to be small enough instead of being convergent to zero as number of training data \( N \to \infty \).
In this section, we discuss the intrinsic Fourier representation constructed for numerical comparisons in Example II and provide more detailed discussion on the numerical results.

We first discuss the construction for the intrinsic Fourier representation of the true conditional density, \( p(x|D) \), defined with respect to the volume form inherited by \( \mathcal{N} \) from the ambient space \( \mathbb{R}^n \) for the system on the torus (49) in Example II. By noticing the embedding (50) of \((\theta, \phi)\) in \( x \equiv (x, y, z) \), we can obtain the following equality,

\[
1 = \int_{\mathcal{N}} p(x|D)\,dV(x) = \int_{[0,2\pi]^2} p(x(\theta, \phi)|D)\,|x_\theta \times x_\phi|\,d\theta d\phi = \int_{[0,2\pi]^2} p_{IC}(\theta, \phi|D)\,d\theta d\phi, \quad (62)
\]

where \( dV(x) = |x_\theta \times x_\phi|\,d\theta d\phi \) is the volume form, \( p_{IC} \) denotes the true conditional density as a function of the intrinsic coordinates, \((\theta, \phi)\). Assuming that \( p_{IC}(\theta, \phi|D) \in L^2([0,2\pi]^2) \), and the relation in (62), we can construct the intrinsic Fourier representation as follows,

\[
\tilde{p}(x|D) = \frac{\hat{p}_{IC}(\theta, \phi|D)}{|x_\theta \times x_\phi|}, \quad (63)
\]

where \( \hat{p}_{IC}(\theta, \phi|D) \) is a RKWHS representation (5) of the conditional density \( p_{IC}(\theta, \phi|D) \) with a set of orthonormal Fourier basis functions \( \psi_k(\theta, \phi) \in L^2([0,2\pi]^2) \). Here, \( \psi_k(\theta, \phi) \) are formed by the tensor product of two sets of orthonormal Fourier basis functions \( \left\{ 1, \sqrt{2}\cos(m\theta) \right\}_{m \in \mathbb{N}^+}, \left\{ \sqrt{2}\sin(m\theta) \right\}_{m \in \mathbb{N}^+} \) and \( \left\{ \sqrt{2}\cos(m\phi) \right\}_{m \in \mathbb{N}^+}, \left\{ \sqrt{2}\sin(m\phi) \right\}_{m \in \mathbb{N}^+} \). Note that for intrinsic Fourier representation, we need to know the embedding (50) and know the data of \((\theta, \phi)\) in intrinsic coordinates for training, which is available for this example. Nevertheless, for Hermite, Cosine, and VBDM representations, we only need to know the observation data \( x \) for training.

The convergence of \( \tilde{p}(x|D) \) to the true density can be explained as follows. For the system (49) in the intrinsic coordinates \((\theta, \phi)\), where \( p_{IC}(\theta, \phi|D) \in L^2([0,2\pi]^2) \) for all parameter \( D \), the statistics \( \text{Var}_{(\theta, \phi)|D}[\psi_k(\theta, \phi)] \) are bounded for all \( D \) and all \( k \in \mathbb{N}^+ \), by the compactness of \([0,2\pi]^2\) and the uniform boundedness of \( \psi_k(\theta, \phi) \) for all \( k \). According to Theorem 1, we can obtain the convergence of the representation \( \tilde{p}_{IC}(\theta, \phi|D) \). Then by noticing the smoothness of \(|x_\theta \times x_\phi|\) on the torus, we can obtain the convergence of the intrinsic Fourier representation \( \tilde{p}(x|D) \) in (63).

Next, we give an intuitive explanation for the reason why in the regime \( d < n \), VBDM representation can provide a good approximation whereas Hermite and Cosine representations cannot. Essentially, the VBDM representation uses basis functions of the weighted Hilbert space of functions defined with respect to a volume form \( V \) that is conformally equivalent to the volume form \( \overline{V} \) that is inherited by the data manifold \( \mathcal{N} \) from the ambient space, \( \mathbb{R}^n \). That is, the weighted Hilbert space, \( L^2(\mathcal{N},\overline{\eta}^{-1}) \), means,

\[
L^2(\mathcal{N},\overline{\eta}^{-1}) = \left\{ f : \int_{\mathcal{N}} |f(x)|^2d\overline{V}(x) < \infty \right\},
\]

where \( d\overline{V}(x) = \overline{\eta}(x)^{-1}dV(x) \) denotes the volume form that is conformally changed by the sampling density \( \overline{\eta} \). We should point out that the key point of the diffusion maps algorithm [7] is to introduce an appropriate normalization to avoid biased in the geometry induced by the sampling density \( \overline{\eta} \) when the data are not
sampled according to the Riemannian metric inherited by \( \mathcal{N} \) from the ambient space \( \mathbb{R}^n \). Furthermore, the orthonormal basis functions of the Hilbert space \( L^2(\mathcal{N}, \pi^{-1}) \) are the eigenfunctions of the adjoint (with respect to \( L^2(\mathcal{N}) \)) of the operator, \( \mathcal{L} = \nabla \log(\eta) \cdot \nabla + \Delta \), that is constructed by the VBDM algorithm. Incidentally, the adjoint operator \( \mathcal{L}^* \) is the Fokker-Planck operator of a gradient system forced by stochastic noises. The point is that this adjoint operator takes density functions of the weighted Hilbert space \( L^2(\mathcal{N}, \pi^{-1}) \). Since the Hilbert space \( L^2(\mathcal{N}, \pi^{-1}) \) is a function space of some Fokker-Planck operator that acts on densities defined with respect to the geometry of data, then representing the conditional density with basis functions of the weighted Hilbert space \( L^2(\mathcal{N}, \pi^{-1}) \) is a natural choice. Thus, the error estimation in Theorem 2 is valid in controlling the error of the estimate.

Next, we will show that the representation of the true density \( p_{\text{EX}} \) in the ambient space \( \mathbb{R}^3 \) is not a function of \( L^2(\mathcal{R}, q^{-1}) \), where for Hermite representation \( \mathcal{R} \) is \( \mathbb{R}^3 \) and \( q \) is a normal distribution, and for Cosine representation \( \mathcal{R} \) is a hyperrectangle containing the torus and \( q \) is a uniform distribution. Recall that the torus is parametrized by:

\[
x(x, y, z) = ((2 + r \sin(\theta)) \cos(\phi), (2 + r \sin(\theta)) \sin(\phi), r \cos(\theta)) ,
\]

where \( \theta, \phi \) are angles which make a full circle, and \( r \) is the radius of the tube known as the minor radius. All observation data are located on the torus with \( r = 1 \). Then, the generalized conditional density \( p(r, \theta, \phi|D) \) in \( (r, \theta, \phi) \) coordinate can be defined using the Dirac delta function as follows,

\[
p(r, \theta, \phi|D) = p_{\text{IC}}(\theta, \phi|D) \delta(r - 1) ,
\]

where \( p_{\text{IC}} \), defined in (62), denotes the conditional density function in the intrinsic coordinate, \((\theta, \phi)\), and \( \delta \) is the Dirac delta function. After coordinate transformation, the density \( p_{\text{EX}} : \mathbb{R}^3 \rightarrow \mathcal{R} \) can be obtained as

\[
p_{\text{EX}}(x|D) = \frac{p(r, \theta, \phi|D)}{J} = \frac{p_{\text{IC}}(\theta, \phi|D) \delta(r - 1)}{J} ,
\]

where \( J \) is the Jacobian determinant \( \det[\partial(x, y, z)/\partial(r, \theta, \phi)] \). Now it can be clearly seen that due to the Dirac delta function \( \delta(r - 1) \) in (66), the density \( p_{\text{EX}}(x|D) \) is no longer in the weighted Hilbert space, \( p_{\text{EX}}(x|D) \notin L^2(\mathcal{R}, q^{-1}) \). Consequently, the error estimation in Theorem 1 becomes invalid in controlling the error of the conditional density.

Numerically, the density \( p_{\text{EX}}(x|D) \) in (66) cannot be well approximated using only finite number of basis functions for Hermite and Cosine representations (5). If only finite number of Hermite or Cosine basis functions are used for representations, typically Gibbs phenomenon can be observed, i.e., the Dirac delta function \( \delta(r - 1) \) in (66) will be approximated by a function having a single tall spike at \( r = 1 \) with some oscillations at two sides along the \( r \) direction. On the other hand, the data-driven basis functions obtained via the diffusion maps algorithm are smooth functions defined on the data manifold \( \mathcal{N} \). Therefore, while the Gibbs phenomenon still occurs in this spectral expansion, it is due to finite truncation in representing a positive smooth functions (densities) on the data manifold, and not due to the singularity that occurs in the ambient direction as in (66).

References

[1] Nachman Aronszajn. Theory of reproducing kernels. *Trans. Amer. Math. Soc.*, 68(3):337–404, 1950.

[2] James L Beck and Siu-Kui Au. Bayesian updating of structural models and reliability using Markov Chain Monte Carlo simulation. *J. Eng. Mech.*, 128(4):380–391, 2002.

[3] Tyrus Berry and John Harlim. Forecasting turbulent modes with nonparametric diffusion models: Learning from noisy data. *Physica D*, 320:57–76, 2016.
[4] Tyrus Berry and John Harlim. Variable bandwidth diffusion kernels. *Appl. Comput. Harmon. Anal.*, 40(1):68–96, 2016.

[5] Tyrus Berry and John Harlim. Correcting biased observation model error in data assimilation. *Mon. Wea. Rev.*, 145(7):2833–2853, 2017.

[6] Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng. *Handbook of Markov Chain Monte Carlo*. CRC press, 2011.

[7] Ronald R Coifman and Stéphane Lafon. Diffusion maps. *Appl. Comput. Harmon. Anal.*, 21(1):5–30, 2006.

[8] M. Dashti and A.M. Stuart. The Bayesian approach to inverse problems. In R. Ghanem, D. Higdon, and H. Owhadi, editors, *Handbook of Uncertainty Quantification*. Springer.

[9] Heikki Haario, Marko Laine, Antonietta Mira, and Eero Saksman. DRAM: efficient adaptive MCMC. *Stat. Comput.*, 16(4):339–354, 2006.

[10] John Harlim. *Data-Driven Computational Methods: Parameter and Operator Estimations*. Cambridge University Press, UK, 2018.

[11] John Harlim and Haizhao Yang. Diffusion forecasting model with basis functions from QR-decomposition. *J. Nonlinear Sci.*, pages 1–26, 2017.

[12] Dave Higdon, Marc Kennedy, James C Cavendish, John A Cafeo, and Robert D Ryne. Combining field data and computer simulations for calibration and prediction. *SIAM J. Sci. Comput.*, 26(2):448–466, 2004.

[13] Janne MJ Huttunen, Jari P Kaipio, and E Somersalo. Approximation errors in nonstationary inverse problems. *Inverse Probl. Imag.*, 1(1):77–93, 2007.

[14] J.P. Kaipio and E. Somersalo. *Statistical and Computational Inverse Problems*. Springer New York, 2005.

[15] Don O Loftsgaarden and Charles P Quesenberry. A nonparametric estimate of a multivariate density function. *Ann. Math. Statist.*, pages 1049–1051, 1965.

[16] E.N. Lorenz. Predictability: a problem partly solved. In *Seminar on Predictability, 4-8 September 1995*, volume 1, pages 1–18, Shinfield Park, Reading, 1995. ECMWF, ECMWF.

[17] YP Mack and Murray Rosenblatt. Multivariate k-nearest neighbor density estimates. *J. Multivar. Anal.*, 9(1):1–15, 1979.

[18] Y. Marzouk and D. Xiu. A stochastic collocation approach to Bayesian inference in inverse problems. *Commun. Comput. Phys.*, 6:826–847, 2009.

[19] Y.M. Marzouk, H.N. Najm, and L.A. Rahn. Stochastic spectral methods for efficient Bayesian solution of inverse problems. *J. Comput. Phys.*, 224(2):560–586, 2007.

[20] Joseph B Nagel and Bruno Sudret. Spectral likelihood expansions for Bayesian inference. *J. Comput. Phys.*, 309:267–294, 2016.

[21] Radford M Neal et al. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2(11), 2011.

[22] Evert J Nyström. Über die praktische Auflösung von Integralgleichungen mit Anwendungen auf Randwertaufgaben. *Acta Math.*, 54(1):185–204, 1930.
[23] Le Song, Kenji Fukumizu, and Arthur Gretton. Kernel embeddings of conditional distributions: A unified kernel framework for nonparametric inference in graphical models. *IEEE Signal Process. Mag.*, 30(4):98–111, 2013.

[24] Le Song, Jonathan Huang, Alex Smola, and Kenji Fukumizu. Hilbert space embeddings of conditional distributions with applications to dynamical systems. In *Proc. 26th Annual Int. Conf. on Machine Learning*, pages 961–968. ACM, 2009.

[25] Timothy John Sullivan. *Introduction to uncertainty quantification*, volume 63. Springer, 2015.