Automatic reproducing kernel and regularization for learning convolution kernels

Haibo Li * and Fei Lu †

Abstract

Learning convolution kernels in operators from data arises in numerous applications and represents an ill-posed inverse problem of broad interest. With scant prior information, kernel methods offer a natural nonparametric approach with regularization. However, a major challenge is to select a proper reproducing kernel, especially as operators and data vary. We show that the input data and convolution operator themselves induce an automatic, data-adaptive RKHS (DA-RKHS), obviating manual kernel selection. In particular, when the observation data is discrete and finite, there is a finite set of automatic basis functions sufficient to represent the estimators in the DA-RKHS, including the minimal-norm least-squares, Tikhonov, and conjugate-gradient estimators. We develop both Tikhonov and scalable iterative and hybrid algorithms using the automatic basis functions. Numerical experiments on integral, nonlocal, and aggregation operators confirm that our automatic RKHS regularization consistently outperforms standard ridge regression and Gaussian process methods with preselected kernels.

Contents

*School of Mathematics and Statistics, The University of Melbourne, Parkville, VIC 3010, Australia. haibo.li@unimelb.edu.au

†Department of Mathematics, Johns Hopkins University, Baltimore, USA. feilu@math.jhu.edu

1 Introduction

Kernel functions play a fundamental role in defining operators between function spaces, enabling the representation of nonlocal or long-range interactions between variables. Such kernel-based operators permeate diverse fields: they describe nonlocal diffusion and peridynamic mechanics in partial differential equations (PDEs) [4, 6, 10, 16, 17, 27, 41, 46], govern anomalous transport in fractional diffusion and Lévy processes [2, 18], and underpin advanced image-processing techniques [5, 22, 34]. More recently, they have become central in operator-learning frameworks for scientific machine learning, from DeepONets [38] and Fourier neural operators [29, 33], nonlocal neural networks [1, 45], and kernel methods [9, 15, 39].

Motivated by these applications, a natural and challenging inverse problem arises: given pairs of inputs and outputs, how can one accurately recover the underlying kernel? We address this question in the linear setting, where the operator acts by convolution against a functional of the input function. By framing kernel recovery as a deconvolution inverse problem, we lay the groundwork for rigorous analysis and practical algorithms that learn these kernels directly from data, bridging the gap between classical inverse problems and modern data-driven operator learning.

1.1 Problem statement

We study the problem of estimating a convolution kernel $\phi : \mathcal{S} = [0,1] \to \mathbb{R}$ in the operator $R_\phi : \mathbb{X} \to \mathbb{Y}$ of the form

$$R_\phi[u](x) = \int_{\mathcal{S}} \phi(s)\, g[u](x,s)\, ds, \quad x \in \mathcal{X} = \{x_j\}_{j=1}^J \subset [0,1], \tag{1.1}$$

based on discrete and noisy input-output pairs

$$\mathcal{D} = \{(u_k(y_i), f_k(x_j)) : \ 1 \leqslant k \leqslant n_0, \ 1 \leqslant i \leqslant 3J, \ 1 \leqslant j \leqslant J\}. \tag{1.2}$$

Here, $\{y_i\}_{i=1}^{3J}$ and $\{x_j\}_{j=1}^J$ are uniform meshes of $[-1,2]$ and $[0,1]$ with mesh sizes $\Delta x = y_{i+1} - y_i = x_{j+1} - x_j = \frac{1}{J}$, and these data are generated according to

$$f_k(x_j) = R_\phi[u_k](x_j) + \epsilon_k(x_j), \quad \epsilon_k(x_j) \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2/\Delta x).$$

The input function space $\mathbb{X} \subset L^2([-1,2])$ and the functional $g[u](x,s)$ are problem-specific (see Examples 1.1– 1.3). Note that R_ϕ can be a nonlinear functional of u, but it depends linearly on ϕ. In this study, we consider a fixed discrete observation set $\mathcal{X}$ and set the output function space to be $\mathbb{Y} = L^2_\nu(\mathcal{X})$ with an atomic measure ν defined by $\nu(\{x_j\}) = 1/J$. When $\mathcal{X}$ is a continuum set $[0,1]$ with Lebesgue measure, the corresponding output space is $L^2([0,1])$, the noise is white, and the minimax convergence rates in the sample size n_0 have been studied in [50].

Such deconvolution-type problems arise in a wide range of applications, and we present three representative examples.

Example 1.1 (Integral operator) *Estimate $\phi : \mathcal{S} \to \mathbb{R}$ in the integral operator*

$$R_\phi[u](x) = \int_{[-1,2]} \phi(x-y)\, u(y)\, dy = \int_{[0,1]} \phi(s)\, u(x-s)\, ds$$

with input space $\mathbb{X} = C([-1,2])$. In the form (1.1), the functional is $g[u](x,s) = u(x-s)$ for $(x,s) \in \mathcal{X} \times \mathcal{S}$. The input u is a sample of the stochastic process $u(y) = \sum_{n=2}^{n_u} X_n \cos(2\pi n y)$ with $n_u \leqslant +\infty$, where the coefficients $\{X_n\}$ are independent Gaussian random variables $N(0, 4\sigma_n^2)$ with $\sum_n n\sigma_n < +\infty$.

Example 1.2 (Nonlocal operator) *Estimate $\phi : \mathcal{S} \to \mathbb{R}$ in the nonlocal operator:*

$$R_\phi[u](x) = \int_{|x'| \leqslant 1} \phi(|x'|)\big(u(x+x') - u(x)\big)\, \nu(dx') = \int_{[0,1]} \phi(s)\, g[u](x,s)\, ds,$$

with input space $\mathbb{X} = C^1([-1,2])$ and $g[u](x,s) = u(x+s) + u(x-s) - 2u(x)$. This operator arises in peridynamics [35, 46, 47] and the Fokker-Planck equation of Lévy processes [2].

Example 1.3 (Aggregation operator) *Consider the* aggregation operator $R_\phi[u] = \nabla\cdot(u\nabla\Phi * u)$ *in the mean-field equation $\partial_t u = \nu\Delta u + \nabla\cdot(u\nabla\Phi * u)$ for interacting particle systems [6, 30]. Let Φ be a radial potential supported on $\mathcal{S}$ and set $\phi = \Phi'$. For $u \in \mathbb{X} = C^1([-1,2])$, one has*

$$R_\phi[u](x) = \int_{\mathbb{R}} \phi(|x'|)\frac{x'}{|x'|}\, \partial_x\big[u(x-x')u(x)\big]\, \nu(dx') = \int_{\mathcal{S}} \phi(s) g[u](x,s)\, ds$$

with $g[u](x,s) = \partial_x[u(x-s)u(x)] - \partial_x[u(x+s)u(x)]$. We consider input functions u to be random probability density functions $u(x) = 1 + \sum_{n=1}^{n_u} \sigma_n\, \zeta_n\, \cos\big(2\pi n\, x\big)$, where $\{\zeta_n\}_{n\geqslant 1}$ are i.i.d. random signs (i.e., $\mathbb{P}(\zeta_n = \pm 1) = \frac{1}{2}$)*, and $\sigma_n > 0$ with $\sum_n n\sigma_n < 1$.*

1.2 Main results: automatic reproducing kernel and regularization

Challenges in learning kernels. Learning convolution kernels from discrete, noisy observations is a severely ill-posed inverse problem: even small data perturbations can induce large estimation errors, making regularization indispensable. Moreover, with scant prior knowledge of the true kernel, a nonparametric framework is necessary, rendering the choice of regularization norm both critical and nontrivial.

Kernel methods are particularly suitable for such inverse problems, as they can non-parametrically approximate the unknown functions with regularization using reproducing kernel Hilbert spaces (RKHS). Hence, they have been widely used in machine learning and inverse problems, dating back from solving the Fredholm equations in [42, 43] and functional linear regression [44, 48] to the recent studies on learning dynamical systems [15, 21], one-shot stochastic differential equations [14], linear responses estimations [49], and solvers for nonlinear PDEs [9] and PDEs on manifolds [26], to name just a few. In particular, the representer theorem reduces the problem with finite data to a finite-dimensional form, enabling efficient computation and feature extraction.

However, a major obstacle in kernel methods is the choice of the reproducing kernel. Standard options, such as Gaussian or Matérn kernels, come with hyperparameters (e.g., bandwidth or smoothness order) that must be carefully tuned. This process is not only computationally expensive but also fails to exploit the specific structure of the inverse problem at hand. In particular, when learning kernels in operators, the variational normal operator may be rank-deficient or possess zero eigenvalues, rendering conventional kernel selection and hyperparameter tuning virtually intractable.

Main results. To overcome this obstacle, we propose an *automatic reproducing kernel* that is defined directly in terms of the data and the forward operator. By incorporating the normal operator from the variational formulation, our kernel automatically adapts to the geometry and spectral properties of the inverse problem. The resulting data-adaptive (DA) RKHS has a closure that is the space in which we can identify the true convolution kernel. Moreover, we use the representer theorem to derive a set of *automatic basis functions* that are adaptive to the finite discrete observations and are sufficient to represent the estimators in the DA-RKHS, including the minimal-norm least-squares, Tikhonov, and conjugate-gradient estimators. These basis functions make mesh-free regression possible and reveal the finite-dimensional nature of the seemingly infinite-dimensional inverse problem of deconvolution.

Building on this theory, we develop two families of regularization algorithms for efficient implementations of the automatic reproducing kernel:

- *Tikhonov methods* based on matrix decomposition for small to medium datasets, with regularization parameters chosen via the L-curve or generalized cross-validation criteria.
- *Iterative and hybrid regularization schemes* that rely solely on matrix-vector products, which are scalable for large datasets.

Notations. Throughout, we use roman letters (e.g., f, u, G) and Greek letters (e.g., ϕ, ξ, λ) to denote functions or scalars, with their meanings clear from context. Boldface symbols (e.g., $\mathbf{G}, \mathbf{c}, \mathbf{x}$) denote vectors or matrices; we write $\mathbf{c} = (c_i)$ or $\mathbf{c}(i)$ for its i-th component. We reserve 0 for the zero function and $\mathbf{0}$ for the zero vector, and denote by $\mathbf{I}_k$ the $k \times k$ identity matrix. For a closed linear subspace $\mathcal{H}$, $P_{\mathcal{H}}$ is the orthogonal projection. Given any linear operator or matrix, $\mathcal{N}(\cdot)$ and $\mathcal{R}(\cdot)$ are its null and range spaces, respectively. Finally, for a bounded linear operator T between Hilbert spaces, T^* denotes its adjoint.

The structure of the paper is as follows. In Section 2, we introduce the automatic reproducing kernel and automatic basis functions, and derive regularized estimators based on Tikhonov and iterative regularization methods. In Section 3 and Section 4, we propose practical algorithms for computing the estimators, including the approximations from discrete data and Tikhonov and iterative regularization algorithms for small and large datasets, respectively. We use three typical examples to illustrate the accuracy and efficiency of our methods in Section 5. The conclusion is in Section 6.

2 Automatic reproducing kernel and basis functions

We first provide a brief review of reproducing-kernel methods for a variational formulation of the inverse problems. Leveraging the variational framework, we then introduce the automatic reproducing kernel. Next, we construct a finite set of automatic basis functions for regression and show that, despite the loss function being minimized over an infinite-dimensional function space, the minimizer actually resides in the finite-dimensional space spanned by these basis functions. In the next section, we build on this continuum analysis to develop practical discrete approximations, paving the way for efficient numerical implementation.

We make the following regularity assumption on data and the operator $R_\phi[u]$ in terms of the bivariate function $g[u]$ throughout this study.

Assumption 2.1 *The functions* $\{g[u_k]\}_{k=1}^{n_0} \subset L^2(\mathcal{X} \times \mathcal{S})$ *is uniformly bounded, i.e.,* $C_g := \max_{1 \leqslant k \leqslant n_0} \sup_{x \in \mathcal{X}, s \in \mathcal{S}} |g[u](x,s)| < \infty$.

2.1 Kernel methods for ill-posed variational inverse problems

We estimate the convolution kernel by a variational approach that minimizes the loss function over a hypothesis space $\mathcal{H}$:

$$\widehat{\phi} = \arg\min_{\phi\in\mathcal{H}} \mathcal{E}_{\mathcal{D}}(\phi), \quad \mathcal{E}_{\mathcal{D}}(\phi) := \frac{1}{n_0} \sum_{k=1,j=1}^{n_0,J} |R_\phi[u_k](x_j) - f_k(x_j)|^2 \Delta x, \tag{2.1}$$

The integral defining $R_\phi[u_k](x_j)$ requires semi-continuum data $\{g[u_k](x_j, s), s \in \mathcal{S}\}_{k,j=1}^{n_0,J}$ that is discrete in x and continuous in s, which in turn presumes access to the continuum data u_k. In practice, however, we only observe discrete data u_k as in (1.2) yielding the values to discrete $\{g[u_k](x_j, s_l); l = 1, \ldots, n_s\}_{k,j=1}^{n_0,J}$. In Section 3, we use these discrete data to empirically approximate the integrals in $R_\phi[u_k](x_j)$.

Two preliminary tasks in this variational approach are to select a hypothesis space $\mathcal{H}$ along with a representation of the function ϕ, and select a penalty term for regularization, which is crucial for the deconvolution-type problem.

Kernel methods achieve both tasks by selecting a reproducing kernel, which provides a reproducing kernel Hilbert space (RKHS) as the hypothesis space and provides an RKHS norm as the penalty term. Specifically, let K be a reproducing kernel (a positive definite function on $\mathcal{S} \times \mathcal{S}$) and denote its RKHS by H_K. Each function in the RKHS can be represented by $\phi(s) = \sum_{l=1}^{n_s} c_l K(s_l, s)$, whose RKHS norm is $\|\phi\|_{H_K} = \sqrt{\sum_{ij} c_i c_j K(s_i, s_j)}$. Here, the sample points $\{s_l\}_{l=1}^{n_s}$ must be properly chosen to extract enough features. Then, the minimizer of the quadratic loss function follows from solving the coefficients $\mathbf{c} = (c_1, \ldots, c_{n_s})$ via least squares with a penalty term depending on $\|\phi\|_{H_K}^2$.

However, the choice of the reproducing kernel K is a major challenge. The widely used reproducing kernels, such as Gaussian kernels, come with hyperparameters that must be carefully tuned along with the regularization strength parameter. This process is not only computationally expensive but also fails to leverage the specific structure of the forward operator $R_\phi[u]$. In particular, when the quadratic loss function is not strictly convex, the conventional kernel selection and hyperparameter tuning are challenging.

We address this challenge in the next section by introducing an automatic reproducing kernel that is adaptive to the data and the forward operator.

2.2 Automatic reproducing kernel and RKHS

We first introduce a weighted function space $L^2_\rho(\mathcal{S})$, where the measure ρ defined below quantifies the exploration of data to the unknown function through the functions $\{g[u_k](x, \cdot)\}_k$, hence it is referred to as an *exploration measure*.

Definition 2.2 *Given data satisfying* Assumption 2.1*, let ρ be a measure on $\mathcal{S}$ with a density function with respect to the Lebesgue measure:*

$$\dot{\rho}(s) := \frac{1}{n_0 Z} \sum_{k=1}^{n_0} \int_{\mathcal{X}} |g[u_k](x, s)| \nu(dx), \ \forall s \in \mathcal{S}, \tag{2.2}$$

where $Z = \frac{1}{n_0} \sum_{k=1}^{n_0} \int_{\mathcal{S}} \int_{\mathcal{X}} |g[u_k](x, s)| \nu(dx) ds$ is the normalization constant.

The exploration measure plays the role of the probability measure ρ_X in nonparametric regression of $f(x) = \mathbb{E}[Y \,|X = x] \in L^2_{\rho_X}(\mathcal{S})$ from data $\{(x_i, y_i)\}$ that are samples of the joint

distribution (X, Y) (see e.g., [12, 24]). Here, we use the L^1 norm of $g[u](\cdot, s)$; alternatively, one can also use the L^2 norm, as in [50], to relax the constraint on $g[u]$. It is particularly useful when treating singular kernels in nonlocal operators, which may not be square integrable with respect to the Lebesgue measure, but square integrable in L^2_ρ. For example, $\phi(s) = s^{-\alpha} \notin L^2([0, \delta])$ for $\alpha \in (\frac{1}{2}, \frac{3}{2})$, but $\phi \in L^2_\rho(\mathcal{S})$ when $u_k \in C^2[0, 1]$ with uniformly bounded second-order derivatives since $\dot{\rho}(s) = O(s^2)$ for small s since $g[u_k](x, s) = u_k(x+s)+u_k(x-s)-2u_k(x) = u_k''(x)s^2/2+o(s^2)$.

Next, we introduce the automatic reproducing kernel.

Definition 2.3 (Automatic reproducing kernel) *The automatic reproducing kernel for estimating ϕ in the operator R_ϕ in (1.1) from data $\{g[u_k](x, \cdot)\}_k$ is the function $\overline{G} : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ defined by*

$$\overline{G}(s, s') := \frac{G(s, s')}{\dot{\rho}(s)\dot{\rho}(s')} \mathbf{1}_{\{\dot{\rho}(s)\dot{\rho}(s')>0\}}, \quad G(s, s') := \frac{1}{n_0} \sum_{k=1}^{n_0} \int_{\mathcal{X}} g[u_k](x, s) g[u_k](x, s') \nu(dx), \tag{2.3}$$

where $\dot{\rho}$ is defined in (2.2).

The next lemma shows that the automatic reproducing kernel comes from the quadratic term of the loss function $\mathcal{E}_\mathcal{D}(\phi)$ in (2.1). In particular, the closure (in L^2_ρ) of its RKHS is the space in which the variational problem has a unique minimizer. Its proof is postponed to Appendix A. For notation simplicity, we write $L^2_\rho(\mathcal{S})$ and $L^2_{\rho\otimes\rho}(\mathcal{S} \times \mathcal{S})$ as L^2_ρ and $L^2_{\rho\otimes\rho}$, respectively.

Lemma 2.4 *Under Assumption 2.1, the following statements hold true:*

(a) $\overline{G}(s, s')$ is in $L^2_{\rho\otimes\rho}$ and symmetric, and the operator $\mathcal{L}_{\overline{G}} : L^2_\rho \to L^2_\rho$ defined by

$$\mathcal{L}_{\overline{G}}\phi(s) := \int_{\mathcal{S}} \phi(s')\overline{G}(s, s')\rho(ds') \tag{2.4}$$

is compact, self-adjoint, and positive. Hence, its eigenvalues $\{\lambda_i\}_{i\geqslant 1}$ are nonnegative and its orthonormal eigenfunctions $\{\psi_i\}_i$ form a complete basis of L^2_ρ.

(b) The loss function $\mathcal{E}_\mathcal{D}(\phi)$ in (2.1) can be written as

$$\mathcal{E}_\mathcal{D}(\phi) = \langle \mathcal{L}_{\overline{G}}\phi, \phi\rangle_{L^2_\rho} - 2\langle \phi^D, \phi\rangle_{L^2_\rho} + const., \tag{2.5}$$

where ϕ^D comes from the Riesz representation, $\langle \phi^D, \phi\rangle_{L^2_\rho} = \frac{1}{n_0}\sum_{k=1}^{n_0}\langle R_\phi[u_k], f_k\rangle_{\mathbb{Y}}$ for any $\phi \in L^2_\rho$. In particular, when the data is noiseless, $\phi^D = \mathcal{L}_{\overline{G}}\phi_*$ and the loss function $\mathcal{E}_\mathcal{D}$ has a unique minimizer $\widehat{\phi} = \mathcal{L}_{\overline{G}}{}^{-1}\phi^D = P_H(\phi_*)$ in $H := \overline{\text{span}\{\psi_i\}_{i:\lambda_i>0}} = \mathcal{N}(\mathcal{L}_{\overline{G}})^\perp \subset L^2_\rho$.

(c) The RKHS of $\overline{G}$ is $H_{\overline{G}} := \mathcal{L}_{\overline{G}}{}^{\frac{1}{2}}(L^2_\rho)$ with inner product $\langle \phi, \phi\rangle_{H_{\overline{G}}} = \langle \mathcal{L}_{\overline{G}}{}^{-\frac{1}{2}}\phi, \mathcal{L}_{\overline{G}}{}^{-\frac{1}{2}}\phi\rangle_{L^2_\rho}$. We have $H = \overline{H_{\overline{G}}}$ with closure in L^2_ρ and $\langle \phi, \mathcal{L}_{\overline{G}}\psi\rangle_{H_{\overline{G}}} = \langle \phi, \psi\rangle_{L^2_\rho}$ for any $\phi \in H_{\overline{G}}, \psi \in L^2_\rho$.

Therefore, solving $\nabla\mathcal{E}(\phi) = 2(\mathcal{L}_{\overline{G}}\phi - \phi^D) = 0$ yields the formal solution $\mathcal{L}_{\overline{G}}{}^{-1}\phi^D$. This inverse exists in H when $\phi^D \in \mathcal{L}_{\overline{G}}(L^2_\rho)$ (and is undefined otherwise). Because $\mathcal{L}_{\overline{G}}$ is compact and may even be rank-deficient, the variational inverse problem is ill-posed, making regularization essential for a stable and accurate approximation of the true ϕ. The RKHS $H_{\overline{G}}$ provides a natural regularization space: its closure is H, and it automatically filters out any component of ϕ^D not in $\mathcal{L}_{\overline{G}}(L^2_\rho)$, which arises solely from noise or model error (see [8, Theorem 2.7] for a decomposition of ϕ^D).

2.3 Automatic basis functions and Tikhonov regularization

In the RKHS $H_{\overline{G}}$, we seek a regularized solution by minimizing

$$\min_{\phi \in H_{\overline{G}}} \mathcal{E}_\lambda(\phi) := \mathcal{E}_{\mathcal{D}}(\phi) + \lambda \|\phi\|^2_{H_{\overline{G}}}, \tag{2.6}$$

where $\mathcal{E}_{\mathcal{D}}$ is given in (2.1). In practice, one must choose a finite set of basis functions to represent elements of $H_{\overline{G}}$. A common practice is to use the reproducing kernel to set a hypothesis space $\mathcal{H} = \operatorname{span}\{\overline{G}(s_j, \cdot)\}_{j=1}^{n_s} \subset H_{\overline{G}} \subset L^2_\rho$, where the $\{s_j\}_{j=1}^{n_s} \subset \mathcal{S}$ are sample points. However, this can introduce bias and may fail to capture key features of the underlying inverse problem (see Remark 2.7).

To overcome these limitations, we construct a finite collection of *automatic basis functions* tailored to the semi-continuum observations $\{g[u_k](x_j, \cdot)\}_{k,j=1}^{n_0,J}$. In particular, we show that even though the minimization of the loss function is taken over an infinite-dimensional space, the minimizer actually lies within the finite-dimensional span of these automatic bases. This result extends the classical finite-dimensional representer theorem for smoothing spline in [44, Theorem 1.3.1] to our data-adaptive setting.

Theorem 2.5 (Finite-dimensional representer) *Given functions $\{g[u_k](x_j, \cdot)\}_{k,j=1}^{n_0,J}$, let*

$$\xi_{kj}(s) = \mathcal{L}_{\overline{G}}[\frac{g[u_k](x_j, \cdot)}{\dot\rho(\cdot)}](s) = \int_{\mathcal{S}} \overline{G}(s, s') g[u_k](x_j, s') ds' \tag{2.7}$$

for each k, j. Then, $\xi_{kj} \in H_{\overline{G}}$ and $\langle \xi_{kj}, \phi\rangle_{H_{\overline{G}}} = R_\phi[u_k](x_j)$. Let

$$\mathbf{\Sigma} = (\langle \xi_{kj}, \xi_{k'j'}\rangle_{H_{\overline{G}}}) \in \mathbb{R}^{n_0 J \times n_0 J}, \quad \mathbf{f} = (f_k(x_j)) \in \mathbb{R}^{n_0 J}. \tag{2.8}$$

We have finite-dimensional representations for the estimators in (2.1) and (2.6) as follows.

(a) The least squares estimator with minimal $H_{\overline{G}}$-norm is

$$\widetilde{\phi} = \operatorname*{arg\,min}_{\psi \in \operatorname*{arg\,min}_{\phi \in H_{\overline{G}}} \mathcal{E}_{\mathcal{D}}(\phi)} \|\psi\|^2_{H_{\overline{G}}} = \sum_{kj} \widetilde{c}_{kj} \xi_{kj}, \quad \text{with } (\widetilde{c}_{kj}) =: \widetilde{\mathbf{c}} = \mathbf{\Sigma}^\dagger \mathbf{f}, \tag{2.9}$$

where $\widetilde{\mathbf{c}}$ is the minimal 2-norm solution of $\min_{\mathbf{c}}\{\frac{1}{n_0 J}\|\mathbf{\Sigma}\mathbf{c} - \mathbf{f}\|_2^2\}$.

(b) The estimator of Tikhonov regularization with $H_{\overline{G}}$-norm and $\lambda > 0$ is

$$\widehat{\phi}_\lambda = \operatorname*{arg\,min}_{\phi \in H_{\overline{G}}} \mathcal{E}_\lambda(\phi) = \sum_{kj} \widehat{c}_{kj} \xi_{kj}, \quad \widehat{\mathbf{c}}_\lambda = (\mathbf{\Sigma}^2 + n_0 J \lambda \mathbf{\Sigma})^\dagger \mathbf{\Sigma} \mathbf{f}, \tag{2.10}$$

where the coefficient $\widehat{\mathbf{c}}_\lambda = (\widehat{c}_{kj}) \in \mathbb{R}^{n_0 J}$ solves $\min_{\mathbf{c}}\{\frac{1}{n_0 J}\|\mathbf{\Sigma}\mathbf{c} - \mathbf{f}\|_2^2 + \lambda \mathbf{c}^\top \mathbf{\Sigma}\mathbf{c}\}$.

Proof. First, since $\tilde{g}_{kj} := \frac{g[u_k](x_j, s')}{\dot\rho(s')} \in L^2_\rho$, we have $\xi_{kj} = \mathcal{L}_{\overline{G}} \tilde{g}_{kj} \in H_{\overline{G}}$.

Next, note that for every $\phi \in H_{\overline{G}}$, Lemma 2.4(c) implies that $\langle \mathcal{L}_{\overline{G}} \psi, \phi\rangle_{H_{\overline{G}}} = \langle \psi, \phi\rangle_{L^2_\rho}$ for any $\psi \in L^2_\rho$. Hence,

$$\langle \xi_{kj}, \phi\rangle_{H_{\overline{G}}} = \langle \mathcal{L}_{\overline{G}} \tilde{g}_{kj}, \phi\rangle_{H_{\overline{G}}} = \langle \tilde{g}_{kj}, \phi\rangle_{L^2_\rho} = R_\phi[u_k](x_j).$$

In other words, ξ_{kj} is a representer of the bounded linear functional $R_\phi[u_k](x_j)$ on $H_{\overline{G}}$.

Also, for any $\phi \in H_{\overline{G}}$, we can write it as

$$\phi = \xi + \sum_{kj} c_{kj}\xi_{kj}, \quad \xi \perp \operatorname{span}\{\xi_{kj}\}_{k,j=1}^{n_0,J}.$$

Then, we have $\|\phi\|_{H_{\overline{G}}}^2 = \mathbf{c}^\top \boldsymbol{\Sigma}\mathbf{c} + \|\xi\|_{H_{\overline{G}}}^2$ and $\langle \xi_{kj}, \phi\rangle_{H_{\overline{G}}} = (\boldsymbol{\Sigma}\mathbf{c})_{kj}$. As a result, the loss functions $\mathcal{E}_{\mathcal{D}}(\phi)$ and $\mathcal{E}_\lambda(\phi)$ can be written as

$$\mathcal{E}_{\mathcal{D}}(\phi) = \frac{1}{n_0 J}\sum_{kj} |f_k(x_j) - \langle \xi_{kj}, \phi\rangle_{H_{\overline{G}}}|^2 = \frac{1}{n_0 J}\|\boldsymbol{\Sigma}\mathbf{c} - \mathbf{f}\|_2^2;$$

$$\mathcal{E}_\lambda(\phi) = \frac{1}{n_0 J}\|\boldsymbol{\Sigma}\mathbf{c} - \mathbf{f}\|_2^2 + \lambda \mathbf{c}^\top \boldsymbol{\Sigma}\mathbf{c} + \lambda\|\xi\|_{H_{\overline{G}}}^2.$$

Therefore, the minimizer $\widetilde{\phi}$ of $\mathcal{E}_{\mathcal{D}}(\phi) = \frac{1}{n_0 J}\|\boldsymbol{\Sigma}\mathbf{c} - \mathbf{f}\|_2^2$ with minimal $H_{\overline{G}}$-norm in (2.9) has a coefficient $\widetilde{\mathbf{c}}$ that can be solved with by the minimizer of $\frac{1}{n_0 J}\|\boldsymbol{\Sigma}\mathbf{c} - \mathbf{f}\|_2^2$ with minimal $\|\mathbf{c}\|_2$.

Also, the minimizer of $\mathcal{E}_\lambda(\phi)$ solves $(\frac{1}{n_0 J}\boldsymbol{\Sigma}^2 + \lambda\boldsymbol{\Sigma})\mathbf{c} = \boldsymbol{\Sigma}\mathbf{f}$, which gives (2.10). ■

Note that the matrix $\boldsymbol{\Sigma}$ in (2.8) can be either singular or non-singular. If it is non-singular, the Tikhonov regularized estimator in (2.10) becomes $\mathbf{c}_{ridge} = (\boldsymbol{\Sigma} + n_0 J\lambda I)^{-1}\mathbf{f}$ after canceling out $\boldsymbol{\Sigma}$, which is the widely used ridge regularized estimator. However, when $\boldsymbol{\Sigma}$ is singular, this Tikhonov regularized estimator is different from the ridge estimator. It is $\mathbf{c}_\lambda = (\boldsymbol{\Sigma}^2 + n_0 J\lambda\boldsymbol{\Sigma})^\dagger\boldsymbol{\Sigma}\mathbf{f} = (\boldsymbol{\Sigma} + n_0 J\lambda I)^{-1}P_{\mathcal{N}(\boldsymbol{\Sigma})^\perp}\mathbf{f}$, which prevents the error in $P_{\mathcal{N}(\boldsymbol{\Sigma})}\mathbf{f}$ from contaminating the estimator. In contrast, the ridge regularized estimator will be contaminated by the error $P_{\mathcal{N}(\boldsymbol{\Sigma})}\mathbf{f}$ and would lead to disastrous results in the small noise limit [8,31]. Additionally, in either case, our Tikhonov solution in (2.10) converges to the least squares solution with the minimal norm as $\lambda \to 0$.

Importantly, a singular $\boldsymbol{\Sigma}$ does not imply multiple minimizers for the loss function over the function space, though it leads to multiple minimizers in the coefficient space. As the next remark shows, all the coefficient minimizers correspond to the same function minimizer because when $\boldsymbol{\Sigma}$ is singular, the basis functions $\{\xi_{kj}\}$ are linearly dependent. In short, the loss function $\mathcal{E}_{\mathcal{D}}$ always has a unique minimizer in $H_{\overline{G}}$ regardless of $\boldsymbol{\Sigma}$ being singular or not.

Remark 2.6 *When $\boldsymbol{\Sigma}$ is singular, there are infinitely many $\mathbf{c}$ minimizing $\frac{1}{n_0 J}\|\boldsymbol{\Sigma}\mathbf{c} - \mathbf{f}\|_2^2$, but all such minimizers lead to the unique minimizer in $H_{\overline{G}}$. Equivalently, the set $\{\phi = \sum_{kj} c_{kj}\xi_{kj} : \mathbf{c} = (c_{kj}) \in \mathcal{C}\}$ contains only one element, where $\mathcal{C} = \widetilde{\mathbf{c}} + \mathcal{N}(\boldsymbol{\Sigma})$ is the set of all minimizers of $\|\boldsymbol{\Sigma}\mathbf{c} - \mathbf{f}\|_2^2$. To see it, let ϕ_1 and ϕ_2 be such two elements with corresponding $\mathbf{c}_1, \mathbf{c}_2 \in \mathcal{C}$. Then $\mathbf{c}_1 - \mathbf{c}_2 \in \mathcal{N}(\boldsymbol{\Sigma})$ and $\phi_1 - \phi_2 = \sum_{kj}(\mathbf{c}_1 - \mathbf{c}_2)(kj)\xi_{kj}$. Therefore, $\|\phi_1 - \phi_2\|_{H_{\overline{G}}}^2 = (\mathbf{c}_1 - \mathbf{c}_2)^\top\boldsymbol{\Sigma}(\mathbf{c}_1 - \mathbf{c}_2) = 0$, leading to $\phi_1 = \phi_2$. The same conclusion holds for the regularized loss $\frac{1}{n_0 J}\|\boldsymbol{\Sigma}\mathbf{c} - \mathbf{f}\|_2^2 + \lambda\mathbf{c}^\top\boldsymbol{\Sigma}\mathbf{c}$. In other words, when $\boldsymbol{\Sigma}$ is singular, the basis functions $\{\xi_{kj}\}$ are linearly dependent, so there are multiple coefficients that represent the same function minimizer.*

Remark 2.7 (Basis functions via the reproducing kernel.) *A default approach to solve (2.6) is to use basis functions of the reproducing kernel $\overline{G}$ since $H_{\overline{G}} = \overline{\operatorname{span}\{\overline{G}(s,\cdot)\}_{s\in\mathcal{S}}}$. That is, take sample points $\{s_l\}_{l=1}^{n_s} \subset \mathcal{S}$ and set hypothesis space to be $\mathcal{H} = \operatorname{span}\{\overline{G}(s_l,\cdot)\}_{l=1}^{n_s}$. For any $\phi(s) = \sum_{l=1}^{n_s} a_l\overline{G}(s_l, s) \in \mathcal{H}$, the square of its RKHS norm is*

$$\begin{aligned}\|\phi\|_{H_{\overline{G}}}^2 &= \|\sum_{l=1}^{n_s} a_l\overline{G}(s_l,\cdot)\|_{H_{\overline{G}}}^2 = \sum_{l,l'=1}^{n_s} a_l a_{l'}\langle\overline{G}(s_l,\cdot), \overline{G}(s_l,\cdot)\rangle_{H_{\overline{G}}} = \mathbf{a}^\top\overline{\mathbf{G}}\mathbf{a},\\ \overline{\mathbf{G}}(l,l') &= \langle\overline{G}(s_l,\cdot), \overline{G}(s_l,\cdot)\rangle_{H_{\overline{G}}} = \overline{G}(s_l, s_{l'}), \quad 1 \leqslant l, l' \leqslant n_s.\end{aligned} \tag{2.11}$$

Then, the minimizer of $\mathcal{E}_{\mathcal{D}}(\phi) + \lambda \|\phi\|^2_{H_{\overline{G}}} = \mathbf{a}^\top \mathbf{A}\mathbf{a} - 2\mathbf{a}^\top \mathbf{b} + Const + \lambda \mathbf{a}^\top \overline{\mathbf{G}}\mathbf{a}$ *is*

$$\begin{aligned} \widehat{\mathbf{a}}_\lambda &= (\mathbf{A} + \lambda \overline{\mathbf{G}})^\dagger \mathbf{b}, \quad \textit{with} \\ \mathbf{A}(l,l') &= \langle \mathcal{L}_{\overline{G}} \overline{G}(s_l, \cdot), \overline{G}(s_{l'}, \cdot) \rangle_{L^2_\rho} = \int_{\mathcal{S}} \int_{\mathcal{S}} \overline{G}(s_l, s) \overline{G}(s_{l'}, s') G(s, s') ds ds', \\ \mathbf{b}(l) &= \frac{1}{n_0} \sum_{k=1}^{n_0} \langle R_{\overline{G}(s_l, \cdot)}[u_k], f \rangle_{\mathbb{Y}} = \int_{\mathcal{S}} \overline{G}(s_l, s) \frac{1}{n_0} \sum_{k=1}^{n_0} \int_{\mathcal{X}} g[u_k](x, s) f_k(x) \nu(dx) ds. \end{aligned} \tag{2.12}$$

The major difficulty is selecting the sample points $\{s_j\}$ *such that the resulting basis functions capture all the features in the data. In practice, this only succeeds when* $\{s_j\}$ *align exactly with the* x*-mesh, at which point the estimator coincides with the automatic basis estimator. By contrast, the automatic basis functions in Theorem 2.5 are guaranteed to extract all the features available in the data. Therefore, throughout this study, we employ the automatic basis functions.*

2.4 Conjugate gradient and iterative regularization

Iterative regularization methods circumvent the computationally expensive matrix inversions or decompositions for Tikhonov regularization in (2.10) by minimizing the loss function on a sequence of growing subspaces with early stopping; see, e.g., [20, Ch. 3.3]. These subspaces are designed to progressively capture the dominant features of the true solution, and early stopping prevents the inclusion of noise-dominated directions.

To design iterative regularization methods adapted to the automatic basis functions, we can apply the conjugate gradient (CG) method (see, e.g., [20, Ch. 7]) to the normal equation of the least squares problem

$$\arg\min_{\phi \in H_{\overline{G}}} n_0 J \mathcal{E}_{\mathcal{D}}(\phi) = \|T\phi - \mathbf{f}\|_2^2, \tag{2.13}$$

where T is the linear operator

$$T : H_{\overline{G}} \to (\mathbb{R}^{n_0 J}, \langle \cdot, \cdot \rangle_2), \quad \phi \mapsto (\langle \xi_{kj}, \phi \rangle_{H_{\overline{G}}}). \tag{2.14}$$

At each iteration, the CG method selects a new search direction that is conjugate with respect to the normal operator T^*T to all previous directions. In particular, CG is essentially a Krylov subspace method, where the solution to (2.13) in the l-th CG iteration with initial guess $\phi_0 = 0$ is

$$\phi_l = \arg\min_{\phi \in \mathcal{H}_l} \|T\phi - \mathbf{f}\|_2, \quad \mathcal{H}_l := \operatorname{span}\{(T^*T)^i T^* \mathbf{f}\}_{i=0}^{l-1}. \tag{2.15}$$

Each $\mathcal{H}_l$ is a subspace of $H_{\overline{G}}$ and we call it the l-th *RKHS-Krylov subspace*.

The following theorem shows the implementation of the above CG iteration in the coefficient space of the automatic basis functions.

Theorem 2.8 (Conjugate gradient solutions) *At the* l*-th iteration, the CG solution in (2.15) is* $\phi_l = \pi(\mathbf{c}) = \sum_{kj} \mathbf{c}_l(kj) \xi_{kj}$ *with*

$$\mathbf{c}_l = \arg\min_{\mathbf{c} \in \mathcal{K}_l} \|\mathbf{\Sigma}\mathbf{c} - \mathbf{f}\|_2, \quad \mathcal{K}_l := \operatorname{span}\{\mathbf{\Sigma}^i \mathbf{\Sigma}^\dagger \mathbf{f}\}_{i=1}^{l}, \tag{2.16}$$

and $\mathcal{H}_l = \pi(\mathcal{K}_l) = \operatorname{span}\{\sum_{kj} \mathbf{a}_i(kj) \xi_{kj}\}_{i=1}^l$*, where* $\mathbf{a}_i = \mathbf{\Sigma}^i \mathbf{\Sigma}^\dagger \mathbf{f}$ *and* π *is the linear operator*

$$\pi : \mathbb{R}^{n_0 J} \to H_0 := \operatorname{span}\{\xi_{kj}\}_{k,j=1}^{n_0, J} \subset H_{\overline{G}}, \quad \mathbf{c} \mapsto \sum_{kj} c_{kj} \xi_{kj}. \tag{2.17}$$

In particular, $T\phi = \mathbf{\Sigma c}$ *for any* $\phi = \sum_{kj} c_{kj}\xi_{kj} + \xi$ *with* $\xi \in H_0^{\perp}$ *and* $\mathbf{c} = (c_{kj}) \in \mathbb{R}^{n_0 J}$, $T \circ \pi = \mathbf{\Sigma}$ *and* $T^* = \pi \circ (\mathbf{\Sigma}^\dagger \mathbf{\Sigma})$, *implying the following two commutative diagrams:*

$$\begin{array}{ccc} H_{\overline{G}} & \xrightarrow{\ T\ } & \mathbb{R}^{n_0 J} \\ {\scriptstyle\pi}\uparrow & \nearrow_{\mathbf{\Sigma}} & \\ \mathbb{R}^{n_0 J} & & \end{array} \qquad \begin{array}{ccc} H_{\overline{G}} & \xleftarrow{\ T^*\ } & \mathbb{R}^{n_0 J} \\ {\scriptstyle\pi}\uparrow & \swarrow_{\mathbf{\Sigma}^\dagger \mathbf{\Sigma}} & \\ \mathbb{R}^{n_0 J} & & \end{array} . \tag{2.18}$$

Proof. The proof includes the following four steps.

Step 1: Prove that $T\phi = \mathbf{\Sigma c}$ for any $\phi = \sum_{kj} c_{kj}\xi_{kj} + \xi$ with $\xi \in H_0^{\perp}$ and $\mathbf{c} = (c_{kj}) \in \mathbb{R}^{n_0 J}$, which implies $T \circ \pi = \mathbf{\Sigma}$. By the definition of T, it follows that $T\xi_{kj} = (\langle \xi_{k'j'}, \xi_{kj} \rangle_{H_{\overline{G}}})$ with $1 \leqslant k' \leqslant n_0$ and $1 \leqslant j' \leqslant J$. Thus, recalling that $\mathbf{\Sigma} = (\langle \xi_{kj}, \xi_{k'j'} \rangle_{H_{\overline{G}}})$, we have $T\phi = T(\sum_{kj} c_{kj}\xi_{kj}) = \sum_{kj} c_{kj} T\xi_{kj} = \mathbf{\Sigma c}$.

Step 2: Show that $T^* = \pi \circ \mathbf{\Sigma}^\dagger \mathbf{\Sigma}$. It suffices to show that for any $\mathbf{y} \in \mathbb{R}^{n_0 J}$, in the decomposition $T^*\mathbf{y} = \sum_{kj} a_{kj}\xi_{kj} + \bar{\xi}$ with $\bar{\xi} \in H_0^{\perp}$, we have $\bar{\xi} = 0$ and $(a_{kj}) =: \mathbf{a} = \mathbf{\Sigma}^\dagger \mathbf{\Sigma y}$. By the adjoint identity $\langle T\phi, \mathbf{y} \rangle_2 = \langle \phi, T^*\mathbf{y} \rangle_{H_{\overline{G}}}$ for any $\phi = \sum_{kj} c_{kj}\xi_{kj} + \xi$, we have

$$\langle \mathbf{\Sigma c}, \mathbf{y} \rangle_2 = \langle \sum_{kj} c_{kj}\xi_{kj} + \xi, \sum_{kj} a_{kj}\xi_{kj} + \bar{\xi} \rangle_{H_{\overline{G}}} \quad \Leftrightarrow \quad \mathbf{c}^\top \mathbf{\Sigma y} = \mathbf{c}^\top \mathbf{\Sigma \bar{a}} + \langle \xi, \bar{\xi} \rangle_{H_{\overline{G}}}$$

for all $\mathbf{c} \in \mathbb{R}^{n_0 J}$ and $\xi \in H_0^{\perp}$. Thus, taking $\mathbf{c} = \mathbf{0}$, we have $\langle \xi, \bar{\xi} \rangle_{H_{\overline{G}}} = 0$ for all $\xi \in H_0^{\perp}$. Combining with $\bar{\xi} \in H_0^{\perp}$, we get $\bar{\xi} = \mathbf{0}$. Then, we have $\mathbf{c}^\top \mathbf{\Sigma y} = \mathbf{c}^\top \mathbf{\Sigma a}$ for all $\mathbf{c} \in \mathbb{R}^{n_0 J}$, resulting in $\mathbf{\Sigma y} = \mathbf{\Sigma a}$, or equivalently, $\mathbf{a} \in \mathbf{\Sigma}^\dagger \mathbf{\Sigma y} + \mathcal{N}(\mathbf{\Sigma})$. But any two choices of $\mathbf{a}$ differing by an element of $\mathcal{N}(\mathbf{\Sigma})$ give the same $\phi \in H_{\overline{G}}$ (see Remark 2.6). Therefore, we can take $\mathbf{a} = \mathbf{\Sigma}^\dagger \mathbf{\Sigma y}$.

Step 3: Compute $(T^*T)^i T^* \mathbf{f}$. Note that $T^* \circ \mathbf{\Sigma} = (\pi \circ \mathbf{\Sigma}^\dagger \mathbf{\Sigma}) \mathbf{\Sigma} = \pi \circ \mathbf{\Sigma}$ since $\mathbf{\Sigma}^\dagger \mathbf{\Sigma} = \mathbf{\Sigma}\mathbf{\Sigma}^\dagger$. Using $T \circ \pi = \mathbf{\Sigma}$, we have

$$(T^*T)^i \circ \pi = (T^*T)^{i-1} \circ T^* \circ \mathbf{\Sigma} = (T^*T)^{i-1} \circ \pi \circ \mathbf{\Sigma} = \cdots = \pi \circ \mathbf{\Sigma}^i.$$

Now we have $T^*\mathbf{f} = \pi(\mathbf{\Sigma}^\dagger \mathbf{\Sigma f}) = \pi(\mathbf{\Sigma}\mathbf{\Sigma}^\dagger \mathbf{f})$, and

$$(T^*T)^i T^* \mathbf{f} = (T^*T)^i \circ \pi(\mathbf{\Sigma}^\dagger \mathbf{\Sigma f}) = \pi(\mathbf{\Sigma}^i \mathbf{\Sigma}^\dagger \mathbf{\Sigma f}) = \pi(\mathbf{\Sigma}^{i+1} \mathbf{\Sigma}^\dagger \mathbf{f}).$$

Therefore, the l-th RKHS-Krylov subspace is

$$\mathcal{H}_l = \operatorname{span}\{(T^*T)^i T^* \mathbf{f}\}_{i=0}^{l-1} = \pi(\operatorname{span}\{\mathbf{\Sigma}^i \mathbf{\Sigma}^\dagger \mathbf{f}\}_{i=1}^{l}) = \pi(\mathcal{K}_l).$$

Step 4: Prove $\phi_l = \pi(\mathbf{c}_l)$ with $\mathbf{c}_l$ in (2.16), i.e., $\mathbf{c}_l = \underset{\mathbf{c} \in \mathcal{K}_l}{\arg\min} \|\mathbf{\Sigma c} - \mathbf{f}\|_2$. From the above results, we have

$$\phi_l = \underset{\phi \in \pi(\mathcal{K}_l)}{\arg\min} \|T\phi - \mathbf{f}\|_2 = \underset{\substack{\phi = \pi(\mathbf{c}) \\ \mathbf{c} \in \mathcal{K}_l}}{\arg\min} \|T \circ \pi(\mathbf{c}) - \mathbf{f}\|_2 = \underset{\substack{\phi = \pi(\mathbf{c}) \\ \mathbf{c} \in \mathcal{K}_l}}{\arg\min} \|\mathbf{\Sigma c} - \mathbf{f}\|_2.$$

It follows that $\phi_l = \pi(\mathbf{c}_l)$. ■

Theorem 2.8 implies that $\mathcal{R}(T) \subset \mathcal{R}(\mathbf{\Sigma})$. Furthermore, noting that $\mathcal{N}(T) = H_0^{\perp}$, we have $\dim(\mathcal{R}(T)) = \dim(H_{\overline{G}}/\mathcal{N}(T)) = \dim(H_0) = \operatorname{rank}(\mathbf{\Sigma})$, leading to $\mathcal{R}(T) = \mathcal{R}(\mathbf{\Sigma})$. Then, we have

$P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f} \in \mathcal{R}(T)$ and $P_{\mathcal{N}(\mathbf{\Sigma})}\mathbf{f} \perp \mathcal{R}(T)$, leading to $\arg\min_{\phi\in H_{\overline{G}}} \|T\phi - \mathbf{f}\|_2 = \arg\min_{\phi\in H_{\overline{G}}} \|T\phi - P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}\|_2$ and

$$\phi_l = \arg\min_{\phi\in\mathcal{H}_l} \|T\phi - P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}\|_2 = \arg\min_{\substack{\phi=\pi(\mathbf{c})\\ \mathbf{c}\in\mathcal{K}_l}} \|\mathbf{\Sigma}\mathbf{c} - P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}\|_2. \tag{2.19}$$

Therefore, we can obtain $\phi_l = \pi(\mathbf{c}_l)$ by computing $\mathbf{c}_l = \arg\min_{\mathbf{c}\in\mathcal{K}_l} \|\mathbf{\Sigma}\mathbf{c} - P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}\|_2$.

In practice, rather than applying CG directly, we use the Golub-Kahan bidiagonalization (GKB) to explicitly construct the solution subspace $\mathcal{K}_l$ and solve (2.19) iteratively. This approach is mathematically equivalent to CG but avoids explicitly forming T^*T, which is more numerically stable, and the convergence of iterates can be further stabilized using the hybrid regularization method; see [7, 28] for more details.

Derivation of the GKB method. The recursive relations of GKB for $\{T, P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}\}$ is given by:

$$\begin{cases} \beta_1\mathbf{p}_1 = P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}, \\ \alpha_i\psi_i = T^*(\mathbf{p}_i) - \beta_i\psi_{i-1}, \\ \beta_{i+1}\mathbf{p}_{i+1} = T(\psi_i) - \alpha_i\mathbf{p}_i, \end{cases} \tag{2.20}$$

where $\psi_0 := 0$, and $\{\alpha_i, \beta_i\}$ are computed such that $\{\psi_i\} \subset H_{\overline{G}}$ and $\{\mathbf{p}_i\} \subset \mathbb{R}^{n_0 J}$ are orthonormal, and $\text{span}\{\psi_i\}_{i=1}^l = \mathcal{H}_l$. Let $\tilde{\pi} = \pi|_{\mathcal{N}(\mathbf{\Sigma})^\perp}$, where π is defined in (2.17). Note that $\tilde{\pi}$ is injective, and the two commutative diagrams in (2.18) still hold by replacing $\mathbb{R}^{n_0 J}$ with $\mathcal{N}(\mathbf{\Sigma})^\perp$. By Theorem 2.8, for any ψ_i, there exist a unique $\mathbf{q}_i \in \mathcal{N}(\mathbf{\Sigma})^\perp$ such that $\tilde{\pi}(\mathbf{q}_i) = \psi_i$, and $\langle\psi_i, \psi_j\rangle_{H_{\overline{G}}} = \mathbf{q}_i^\top\mathbf{\Sigma}\mathbf{q}_j$. Using $T^*u_i = \tilde{\pi}(\mathbf{\Sigma}^\dagger\mathbf{\Sigma}u_i)$, we have

$$\begin{cases} \alpha_i\tilde{\pi}(\mathbf{q}_i) = \tilde{\pi}(\mathbf{\Sigma}^\dagger\mathbf{\Sigma}\mathbf{p}_i) - \beta_i\tilde{\pi}(q_{i-1}), \\ \beta_{i+1}\mathbf{p}_{i+1} = T\circ\tilde{\pi}(\psi_i) - \alpha_i\mathbf{p}_i. \end{cases}$$

Using $T\circ\tilde{\pi} = \mathbf{\Sigma}$, we obtain the practical GKB recursive relations:

$$\begin{cases} \beta_1\mathbf{p}_1 = P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}, \\ \alpha_i\mathbf{q}_i = \mathbf{p}_i - \beta_i\mathbf{q}_{i-1}, \\ \beta_{i+1}\mathbf{p}_{i+1} = \mathbf{\Sigma}\mathbf{q}_i - \alpha_i\mathbf{p}_i, \end{cases} \tag{2.21}$$

where $q_0 := \mathbf{0}$, and $\{\alpha_i, \beta_i\}$ are computed such that $\{\mathbf{q}_i\} \subset \mathcal{N}(\mathbf{\Sigma})^\perp$ and $\{\mathbf{p}_i\}$ are $\mathbf{\Sigma}$-orthonormal and 2-orthonormal, respectively. Here, to get the second relation of (2.21), we have used $\mathbf{p}_i \in \mathcal{N}(\mathbf{\Sigma})^\perp$, which can be verified by mathematical induction.

Note that $(\mathcal{N}(\mathbf{\Sigma})^\perp, \langle\cdot,\cdot\rangle_{\mathbf{\Sigma}})$ is a finite-dimensional Hilbert space with inner product $\langle\mathbf{x}, \mathbf{x}'\rangle_{\mathbf{\Sigma}} := \mathbf{x}^\top\mathbf{\Sigma}\mathbf{x}$. The following theorem shows that GKB iteratively constructs a $\mathbf{\Sigma}$-orthonormal basis of $\mathcal{K}_l$ in $(\mathcal{N}(\mathbf{\Sigma})^\perp, \langle\cdot,\cdot\rangle_{\mathbf{\Sigma}})$. The proof is in Appendix A.

Theorem 2.9 *Following the notations in Theorem 2.8, define the linear operator:*

$$\widetilde{T} : (\mathcal{N}(\mathbf{\Sigma})^\perp, \langle\cdot,\cdot\rangle_{\mathbf{\Sigma}}) \to (\mathcal{N}(\mathbf{\Sigma})^\perp, \langle\cdot,\cdot\rangle_2), \quad \mathbf{x} \mapsto \mathbf{\Sigma}\mathbf{x}. \tag{2.22}$$

Then (2.21) is the recursive relations of the GKB for $\{\widetilde{T}, P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}\}$, and the following properties hold:

(a) The two groups of vectors $\{\mathbf{q}_i\}_{i=1}^{l}$ *and* $\{\mathbf{p}_i\}_{i=1}^{l}$ *are* $\boldsymbol{\Sigma}$*-orthonormal and 2-orthonormal bases of the Krylov subspace*

$$\mathcal{K}_l(\boldsymbol{\Sigma}, P_{\mathcal{N}(\boldsymbol{\Sigma})^\perp}\mathbf{f}) := \mathrm{span}\{\boldsymbol{\Sigma}^i P_{\mathcal{N}(\boldsymbol{\Sigma})^\perp}\mathbf{f}\}_{i=0}^{l-1} = \mathcal{K}_l. \tag{2.23}$$

(b) Let the terminate step of GKB be $l_t = \arg\min_{i\geqslant 1} \{\alpha_{i+1}\beta_{i+1} = 0\}$*. Then* $l_t \leqslant \mathrm{rank}(\boldsymbol{\Sigma})$*, and* $\phi_{l_t} = \widetilde{\phi}$*, the LS estimator with minimal* $H_{\overline{G}}$*-norm defined in (2.9).*

(c) Let the residual be $\mathbf{r}_l = T\phi_l - \mathbf{f}$*. Then* $\|\mathbf{r}_l\|_2 = \|\boldsymbol{\Sigma}\mathbf{c}_l - \mathbf{f}\|_2$ *and* $\|\phi_l\|_{H_{\overline{G}}} = \|\mathbf{c}_l\|_{\boldsymbol{\Sigma}}$*, and* $\{\|\mathbf{r}_l\|_2\}$ *and* $\{\|\mathbf{c}_l\|_{\boldsymbol{\Sigma}}\}$ *monotonically decreases and increases, respectively.*

For $l \leqslant l_t$, detnote $\mathbf{P}_l = (\mathbf{p}_1, \ldots, \mathbf{p}_{l+1}) \in \mathbb{R}^{n_0 J\times(l+1)}$ and $\mathbf{Q}_l = (\mathbf{q}_1, \ldots, \mathbf{q}_l) \in \mathbb{R}^{n_0 J\times l}$. From (2.21) we have

$$\begin{cases} \beta_1\mathbf{P}_{l+1}\mathbf{e}_1 = P_{\mathcal{N}(\boldsymbol{\Sigma})^\perp}\mathbf{f}, \\ \boldsymbol{\Sigma}\mathbf{Q}_l = \mathbf{P}_{l+1}\mathbf{B}_l, \\ \mathbf{P}_{l+1} = \mathbf{Q}_l\mathbf{B}_l^\top + \alpha_{l+1}\mathbf{q}_{l+1}\mathbf{e}_{l+1}^\top, \end{cases} \tag{2.24}$$

where $\mathbf{e}_1$ and $\mathbf{e}_{l+1}$ are the first and $(l+1)$-th columns of $\mathbf{I}_{l+1}$, and

$$\mathbf{B}_l = \begin{pmatrix} \alpha_1 & & & \\ \beta_2 & \alpha_2 & & \\ & \beta_3 & \ddots & \\ & & \ddots & \alpha_l \\ & & & \beta_{l+1} \end{pmatrix} \in \mathbb{R}^{(l+1)\times l} \tag{2.25}$$

has full column rank. Using Theorem 2.9 and for any $\mathbf{c} \in \mathcal{K}_l$ letting $\mathbf{c} = \mathbf{Q}_l\mathbf{y}$ with $\mathbf{y} \in \mathbb{R}^l$, we have

$$\begin{aligned} \min_{\mathbf{c}\in\mathcal{K}_l} \|\boldsymbol{\Sigma}\mathbf{c} - P_{\mathcal{N}(\boldsymbol{\Sigma})^\perp}\mathbf{f}\|_2 &= \min_{\mathbf{y}\in\mathbb{R}^l} \|\boldsymbol{\Sigma}\mathbf{Q}_l\mathbf{y} - \beta_1\mathbf{P}_{l+1}\mathbf{e}_1\|_2 \\ &= \min_{\mathbf{y}\in\mathbb{R}^l} \|\mathbf{P}_{l+1}(\mathbf{B}_l\mathbf{y} - \beta_1\mathbf{e}_1)\|_2 = \min_{\mathbf{y}\in\mathbb{R}^l} \|\mathbf{B}_l\mathbf{y} - \beta_1\mathbf{e}_1\|_2. \end{aligned}$$

Therefore, the l-th CG solution equals to

$$\phi_l = \pi(\mathbf{c}_l), \quad \mathbf{c}_l = \mathbf{Q}_l\mathbf{y}_l, \quad \mathbf{y}_l = \arg\min_{\mathbf{y}\in\mathbb{R}^l} \|\mathbf{B}_l\mathbf{y} - \beta_1\mathbf{e}_1\|_2. \tag{2.26}$$

In other words, we only need to solve an l-dimensional least squares problem at the l-th iteration to get the coefficient vector.

Early stopping criterion. The CG iteration yields a regularized solution by early stopping. If the noise norm $\|\epsilon\|_2$ is available, where $\epsilon = (\epsilon_k(x_j)) \in \mathbb{R}^{n_0 J}$, then the discrepancy principle (DP) [20] can be used to halt iteration at the earliest instance of l that satisfies

$$\|T\phi_l - \mathbf{f}\|_2 = \|\boldsymbol{\Sigma}\mathbf{c}_l - \mathbf{f}\|_2 \leqslant \tau\|\epsilon\|_2, \tag{2.27}$$

where τ is chosen to be marginally greater than 1. When $\|\epsilon\|_2$ is unavailable, we adopt the L-curve criterion [25], which estimates the ideal early stopping iteration at the corner of the curve represented by

$$\left(\log \|T\phi_l - \mathbf{f}\|_2, \log \|\phi_l\|_{H_{\overline{G}}}\right) = \left(\log \|\boldsymbol{\Sigma}\mathbf{c}_l - \mathbf{f}\|_2, \log \|\mathbf{c}_l\|_{\boldsymbol{\Sigma}}\right). \tag{2.28}$$

Note from Theorem 2.9 that the residual norm decreases monotonically while the solution norm increases monotonically, which together make the "L"-shape of (2.28) possible. For the L-curve method, one must proceed a few iterations beyond the optimal l to find its corner.

Hybrid regularization method. For the iterative method, the regularized solution is sensitive to the iteration number, and the DP or L-curve criterion may yield a suboptimal iteration number, resulting in an over- or under-regularized solution. To stabilize the convergence, we follow the idea of the hybrid regularization method; see e.g. [11, 28]. At each iteration, instead of solving (2.19), we add an $H_{\overline{G}}$-norm Tikhonov regularization term and solve the problem

$$\phi_{\lambda_l} = \arg\min_{\phi\in\mathcal{H}_l} \|T\phi - P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}\|_2 + \lambda_l\|\phi\|^2_{H_{\overline{G}}}, \tag{2.29}$$

where λ_l is the regularization parameter that is updated at each iteration. For any $\phi \in \mathcal{H}_l$, using $\mathbf{B}_l$ and $\mathbf{Q}_l$ in (2.24)–(2.25) and letting $\phi = \pi(\mathbf{c}) = \pi(\mathbf{Q}_l\mathbf{y})$ with $\mathbf{y} \in \mathbb{R}^l$, we obtain

$$\begin{aligned}\min_{\phi\in\mathcal{H}_l}\{\|T\phi - P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}\|_2^2 + \lambda_l\|\phi\|^2_{H_{\overline{G}}}\} &= \min_{\mathbf{c}\in\mathcal{K}_l}\{\|\mathbf{\Sigma}\mathbf{c} - P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}\|_2^2 + \lambda_l\|\mathbf{c}\|^2_{\mathbf{\Sigma}}\}\\ &= \min_{\mathbf{y}\in\mathbb{R}^l}\{\|\mathbf{B}_l\mathbf{y} - \beta_1\mathbf{e}_1\|_2^2 + \lambda_l\|\mathbf{y}\|_2^2\},\end{aligned}$$

where we have used $\mathbf{Q}_l^\top\mathbf{\Sigma}\mathbf{Q}_l = \mathbf{I}_l$. Then, the l-th hybrid solution is

$$\phi_{\lambda_l} = \pi(\mathbf{Q}_l\mathbf{y}_{\lambda_l}), \quad \mathbf{y}_{\lambda_l} = \arg\min_{y\in\mathbb{R}^l}\{\|\mathbf{B}_l\mathbf{y} - \beta_1\mathbf{e}_1\|_2^2 + \lambda_l\|\mathbf{y}\|_2^2\}. \tag{2.30}$$

Therefore, at each step we only need to update λ_l and compute $\mathbf{y}_{\lambda_l}$, which is computationally efficient. We update λ_l by the weighted GCV (WGCV) method; see [11, 32] for more details.

3 Approximation from discrete data in practice

In practice, the data are discrete, as in (1.2). Thus, to apply the automatic reproducing kernel, we need to numerically approximate the integrals in the exploration measure ρ, the automatic reproducing kernel and basis functions, and the matrix $\mathbf{\Sigma}$ in Section 2.2.

The starting point is to approximate the function $\{g[u_k](x_j,\cdot)\}_{k,j=1}^{n_0,J}$. Note that the explicit forms of these functions are unavailable since the data only provides $\{u_k(y_i)\}_{i=1}^{3J}$, values of these functions at finitely many points, but the functions u_k are unknown. For the operators in Examples 1.1–1.3, the data only defines $g[u_k](x_j, s_l)$ with $s_l = l/J$ for $1 \leqslant l \leqslant J$, and one may use a rougher mesh for s than these J points. For generality, let the mesh points for s be $\{s_l\}_{l=1}^{n_s}$. We denote values of the function $\{g[u_k](x_j,\cdot)\}_{k,j=1}^{n_0,J}$ at these mesh points by a vector

$$\mathbf{g}_{kj} = (g[u_k](x_j, s_l))_{l\leqslant n_s} \in \mathbb{R}^{1\times n_s}, \quad 1 \leqslant k \leqslant n_0, 1 \leqslant j \leqslant J.$$

The functions $\{g[u_k](x_j,\cdot)\}$ can then be approximated by various approaches, such as splines, wavelets, or Fourier series. For simplicity, we consider piece-wise constant approximations, i.e.,

$$\widehat{g}_{kj}(s) = \sum_{l=1}^{n_s} g[u_k](x_j, s_l)\mathbf{1}_{I_l}(s), \tag{3.1}$$

with $I_l = (s_{l-1}, s_l]$ with $s_0 = 0$.

Correspondingly, we use the Riemann sum to approximate the integrals of ρ in (2.2), $\overline{G}$ in (2.3), and ξ_{kj} in (2.7). Table 1 presents their approximations using semi-continuum and

discrete data. The density of ρ with the semi-continuum data is $\dot{\rho}(s)\propto\frac{1}{n_0 J}\sum_{kj}|g[u_k](x_j,s)|$, whose approximation is

$$\begin{aligned}\widehat{\dot{\rho}}(s)&\propto\frac{1}{n_0 J}\sum_{kjl}|g[u_k](x_j,s_l)|\mathbf{1}_{I_l}(s)=\frac{1}{n_0 J}\sum_{kj}|\widehat{g}_{kj}(s)|,\\ \Rightarrow\boldsymbol{\rho}_{_D}&\propto\sum_{kj}|\mathbf{g}_{kj}|\in\mathbb{R}^{1\times n_s}.\end{aligned}\tag{3.2}$$

Here, the factor $\frac{1}{J}=|\Delta x|$ since the x-grid is uniform. Note that $\boldsymbol{\rho}_{_D}$ is a discrete representation of the probability measure ρ, and it assigns weights to the sample points $\{s_l\}$. Since we use piece-wise constant approximations, these weights are the probability of ρ on the sets $\{I_l\}$.

Similarly, we approximate the integral kernel $G(s,s'):=\frac{1}{n_0 J}\sum_{kj}g[u_k](x_j,s)g[u_k](x_j,s')$ in (2.3) by

$$\begin{aligned}\widehat{G}_{_D}(s,s'):&=\frac{1}{n_0 J}\sum_{k,j,l,l'}g_{kj}(s_l)g_{kj}(s_{l'})\mathbf{1}_{I_l}(s)\mathbf{1}_{I_{l'}}(s)=\frac{1}{n_0 J}\sum_{kj}\widehat{g}_{kj}(s)\widehat{g}_{kj}(s'),\\ \Rightarrow\mathbf{G}_{_D}&=\frac{1}{n_0 J}\sum_{kj}\mathbf{g}_{kj}^\top\mathbf{g}_{kj}=\frac{1}{n_0 J}\mathbf{g}^\top\mathbf{g}\in\mathbb{R}^{n_s\times n_s}.\end{aligned}$$

Then, $\widehat{\overline{G}}_{_D}$ and $\overline{\mathbf{G}}$ follows directly from the above approximations of ρ and G, as in Table 1.

Lastly, each automatic basis functions $\xi_{kj}=\int_{\mathcal{S}}\overline{G}_{_D}(s,s')g[u_k](x_j,s')ds'$ in (2.7) has approximations and discrete representations based on $\overline{G}$ and $\overline{\mathbf{G}}_{_D}$ as follows,

$$\begin{aligned}\widehat{\xi_{kj}^{D}}(s)&=\frac{1}{n_0 J}\sum_{k',j',l,l'}g_{k'j'}(s_l)g_{k'j'}(s_{l'})g_{kj}(s_{l'})|\Delta s|\mathbf{1}_{I_l}(s),\\ \Rightarrow\boldsymbol{\xi}_{kj}&=\mathbf{g}_{kj}\overline{\mathbf{G}}_{_D}|\Delta s|\in\mathbb{R}^{1\times n_s}.\end{aligned}\tag{3.3}$$

To approximate the normal matrix $\boldsymbol{\Sigma}=(\langle\xi_{kj},\xi_{k'j'}\rangle_{H_{\overline{G}}})\in\mathbb{R}^{n_0J\times n_0J}$, recall that Lemma 2.4(c) implies $\langle\mathcal{L}_{\overline{G}}\psi,\phi\rangle_{H_{\overline{G}}}=\langle\psi,\phi\rangle_{L^2_\rho}$ for any $\psi\in L^2_\rho$. Then, we obtain from (2.7) that

$$\begin{aligned}\langle\xi_{kj},\xi_{k'j'}\rangle_{H_{\overline{G}}}&=\langle\mathcal{L}_{\overline{G}}{}^{-1}\xi_{kj},\xi_{k'j'}\rangle_{L^2_\rho}=\langle\frac{g[u_k](x_j,\cdot)}{\dot{\rho}(\cdot)},\xi_{k'j'}\rangle_{L^2_\rho}\\ &=\int_{\mathcal{S}}g[u_k](x_j,s)\xi_{k'j'}(s)ds\approx\int_{\mathcal{S}}g[u_k](x_j,s)\widehat{\xi_{k'j'}^{D}}(s)ds\approx\mathbf{g}_{kj}\boldsymbol{\xi}_{k'j'}^\top\Delta s.\end{aligned}$$

In other words, the discrete representation of $\boldsymbol{\Sigma}$ is

$$\boldsymbol{\Sigma}_{_D}:=\mathbf{g}\boldsymbol{\xi}^\top\Delta s,\qquad\text{where }\mathbf{g}=(\mathbf{g}_{kj})\in\mathbb{R}^{n_0J\times n_s},\ \boldsymbol{\xi}=(\boldsymbol{\xi}_{kj})\in\mathbb{R}^{n_0J\times n_s}.$$

To conclude, our estimator (2.10) in computational practice is

$$\widehat{\phi}_\lambda=\sum_{kj}\widehat{c}_{kj}\widehat{\xi}_{kj}^{D}=\sum_{l=1}^{n_s}\widehat{\boldsymbol{\phi}}_\lambda(l)\mathbf{1}_{I_l}(s)\Leftrightarrow\widehat{\boldsymbol{\phi}}_\lambda=\widehat{\mathbf{c}}_\lambda^\top\boldsymbol{\xi}\ \text{ with }\widehat{\mathbf{c}}_\lambda=(\boldsymbol{\Sigma}_{_D}^2+n_0J\lambda\boldsymbol{\Sigma}_{_D})^\dagger\boldsymbol{\Sigma}_{_D}\mathbf{f},\tag{3.4}$$

where the basis functions $\{\mathbf{1}_{I_l}(s)\}_{l=1}^{n_s}$ originate from the piecewise constant approximation of the functions $\{g[u_k](x_j,\cdot)\}$ in (3.1).

Table 1: Functions and arrays from the semi-continuum and discrete data.

Semi-continuum Data	Discrete Data	Vector/Arrays
$g_{kj}(s) := g[u_k](x_j, s)$	$\widehat{g}_{kj}(s) = \sum_{l=1}^{n_s} g_{kjl}\mathbf{1}_{I_l}(s)$, with $g_{kjl} := g[u_k](x_j, s_l)$	$\mathbf{g}_{kj} = (g_{kjl})_{1\leqslant l\leqslant n_s} \in \mathbb{R}^{1\times n_s}$ $\mathbf{g} = (\mathbf{g}_{kj}) \in \mathbb{R}^{n_0 J\times n_s}$
$f_k(x)$		$\mathbf{f} = (f_k(x_j)) \in \mathbb{R}^{n_0 J\times 1}$
$\dot{\rho}(s)\propto \sum_{kj} \lvert g_{kj}(s)\rvert \Delta x$	$\widehat{\dot{\rho}}_{D}(s)\propto \sum_{kjl} \lvert g_{kjl}\rvert \Delta x \mathbf{1}_{I_l}(s)$	$\boldsymbol{\rho}_{D}\propto \sum_{kj} \lvert \mathbf{g}_{kj}\rvert \in \mathbb{R}^{1\times n_s}$
$G(s,s') = \frac{1}{n_0 J}\sum_{kj} g_{kj}(s)g_{kj}(s')$	$\widehat{G}_{D}(s,s')$	$\mathbf{G}_{D} = \frac{1}{n_0 J}\mathbf{g}^\top\mathbf{g} \in \mathbb{R}^{n_s\times n_s}$
$\overline{G}(s,s') = \frac{G(s,s')}{\dot{\rho}(s)\dot{\rho}(s')}$	$\widehat{\overline{G}}_{D}(s,s')$	$\overline{\mathbf{G}}_{D} = \frac{\mathbf{G}_{D}}{\boldsymbol{\rho}_{D}^\top\boldsymbol{\rho}_{D}} \in \mathbb{R}^{n_s\times n_s}$
$\xi_{kj}(s) = \int \overline{G}(s,s')g_{kj}(s')ds'$	$\widehat{\xi}_{kj}^{D}(s)$ $= \sum_{l=1}^{n_s} \boldsymbol{\xi}_{kj}(l)\mathbf{1}_{I_l}(s)$	$\boldsymbol{\xi}_{kj} = \mathbf{g}_{kj}\overline{\mathbf{G}}_{D}\lvert\Delta s\rvert \in \mathbb{R}^{1\times n_s}$ $\boldsymbol{\xi} = \mathbf{g}\overline{\mathbf{G}}_{D}\lvert\Delta s\rvert \in \mathbb{R}^{n_0 J\times n_s}$
$\boldsymbol{\Sigma} = (\langle\xi_{kj}, \xi_{k'j'}\rangle_{H_{\overline{G}}})$		$\boldsymbol{\Sigma}_{D} = \mathbf{g}\boldsymbol{\xi}^\top\lvert\Delta s\rvert \in \mathbb{R}^{n_0 J\times n_0 J}$
$\mathcal{E}_{\mathcal{D}}(\phi)$	$\widehat{\mathcal{E}_{\mathcal{D}}}(\phi) = \widehat{\mathcal{E}_{\mathcal{D}}}(\mathbf{c}) = \frac{1}{n_0 J}\lVert\boldsymbol{\Sigma}_{D}\mathbf{c} - \mathbf{f}\rVert^2$	
LSE mini-norm	$\widetilde{\phi} = \sum_{kj} \widetilde{c}_{kj}\widehat{\xi}_{kj}^{D} \Leftrightarrow \quad \widetilde{\boldsymbol{\phi}} = \widetilde{\mathbf{c}}^\top\boldsymbol{\xi}$ with $\widetilde{\mathbf{c}} = \boldsymbol{\Sigma}_{D}^{\dagger}\mathbf{f}$	
Tikhonov Est.	$\widehat{\phi}_\lambda = \sum_{kj} \widehat{c}_{kj,\lambda}\widehat{\xi}_{kj}^{D} \Leftrightarrow \quad \widehat{\boldsymbol{\phi}}_\lambda = \widehat{\mathbf{c}}_\lambda^\top\boldsymbol{\xi}$ with $\widehat{\mathbf{c}}_\lambda = (\boldsymbol{\Sigma}_{D}^2 + n_0 J\lambda\boldsymbol{\Sigma}_{D})^{\dagger}\boldsymbol{\Sigma}_{D}\mathbf{f}$	

We summarize the above approximations from discrete data in Table 1.

Note that $\boldsymbol{\Sigma}_{D}$ is singular when $n_s < n_0 J$. In other words, when estimating ϕ at n_s evaluation points, the number of necessary features (basis functions) is at most n_s, so the $n_0 J$ data-deduced basis functions must be linearly dependent. Consequently, in this case, it is important to not use the ridge estimator $\mathbf{c}_{ridge} = (\boldsymbol{\Sigma}_{D} + n_0 J\lambda I)^{-1}\mathbf{f}$ but use $\widehat{\mathbf{c}} = (\boldsymbol{\Sigma}_{D}^2 + n_0 J\lambda\boldsymbol{\Sigma}_{D})^{\dagger}\boldsymbol{\Sigma}_{D}\mathbf{f}$ instead.

Importantly, the automatic basis functions $\{\xi_{kj}\}$ have two major advantages over the piecewise constants $\{\mathbf{1}_{I_l}(s)\}_{l=1}^{n_s}$ and other spline basis functions. First, if the analytical form of the functions $\{g[u_k](x_j,\cdot)\}$ is given, we can use the automatic basis functions directly with the coefficient $\widehat{\mathbf{c}}_\lambda$ in (3.4). Second, they overcome the difficulty in computing the RKHS norm. For example, if we write $\phi(s) = \sum_l \boldsymbol{\phi}(l)\mathbf{1}_{I_l}(s)$, then the regularized problem becomes $\min_{\boldsymbol{\phi}} \lVert\mathbf{g}\boldsymbol{\phi} - \mathbf{f}\rVert_2^2 + \lambda\lVert\boldsymbol{\phi}\rVert_{C_{rkhs}}^2$, and a major difficulty is to compute the Gram matrix $C_{rkhs} = \big(\langle\mathbf{1}_{I_l}, \mathbf{1}_{I_{l'}}\rangle_{H_{\overline{G}}}\big)_{1\leqslant l,l'\leqslant n_s}$. In contrast, the Gram matrix $\boldsymbol{\Sigma}_{D}$ for the automatic basis functions is directly available.

4 Practical algorithms for computing the estimators

When $n_0 J$ is not large, e.g., up to a few thousands, one can compute the Tikhonov regularized estimator $\widehat{\mathbf{c}}_\lambda = (\boldsymbol{\Sigma}_{D}^2 + n_0 J\lambda\boldsymbol{\Sigma}_{D})^{\dagger}\boldsymbol{\Sigma}_{D}\mathbf{f}$ based on matrix decomposition. When $n_0 J$ is large, the iterative methods can efficiently compute regularized solutions.

4.1 Tikhonov regularization for small datasets

In Tikhonov regularization, we first compute the eigenvalue decomposition: $\boldsymbol{\Sigma}_{D} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top$ with $\mathbf{U} = (\mathbf{u}_1, \ldots, \mathbf{u}_{n_0 J})$ and $\boldsymbol{\Lambda} = \mathrm{diag}(\{\lambda_i\})$, where $\lambda_1 \geqslant \cdots \geqslant \lambda_{n_0 J}$ are the eigenvalues of $\boldsymbol{\Sigma}_{D}$ and $\{\mathbf{u}_i\}_{i\geqslant 1}$ are the corresponding orthonormal eigenvectors. The solution $\widehat{\mathbf{c}}_\lambda = (\boldsymbol{\Sigma}_{D}^2 + n_0 J\lambda\boldsymbol{\Sigma}_{D})^{\dagger}\boldsymbol{\Sigma}_{D}\mathbf{f}$ can be written as $\widehat{\mathbf{c}}_\lambda = \sum_{\lambda_i>0} \mathbf{u}_i\frac{\mathbf{u}_i^\top\mathbf{f}}{\lambda_i + n_0 J\lambda}$. Note that the components $\mathbf{u}_i^\top\mathbf{f}$ corresponding to $\lambda_i = 0$ do not enter the estimator. To handle the numerical rank-deficient of $\boldsymbol{\Sigma}_{D}$ in practical computations, we set a small threshold $\mathtt{tol} > 0$ (e.g., $\mathtt{tol} = 10^{-14}$ for machine precision on the order of 10^{-16})

and let $r = \#\{\lambda_i : \lambda_i > \texttt{tol}\}$ be the numerical rank of $\mathbf{\Sigma}_D$. Then, we compute a regularized $\widehat{\mathbf{c}}_\lambda \in \text{span}\{\mathbf{u}_1, \ldots, \mathbf{u}_r\}$ as follows. Let $\mathbf{U}_r = (\mathbf{u}_1, \ldots, \mathbf{u}_r)$ and $\mathbf{\Lambda} = \text{diag}(\lambda_1, \ldots, \lambda_r)$. With $\mathbf{c} = \mathbf{U}_r\mathbf{y}$ and $\mathbf{\Lambda}_r^{1/2}\mathbf{y} = \mathbf{z}$, the regularization problem $\min_{\mathbf{c}\in\mathcal{R}(\mathbf{\Sigma}_D)}\{\|\mathbf{\Sigma}_D\mathbf{c} - \mathbf{f}\|_2^2 + \lambda\mathbf{c}^\top\mathbf{\Sigma}_D\mathbf{c}\}$ becomes

$$\min_{\mathbf{y}\in\mathbb{R}^r}\{\|\mathbf{U}_r\mathbf{\Lambda}_r\mathbf{y} - \mathbf{f}\|_2^2 + \lambda\|\mathbf{\Lambda}_r^{1/2}\mathbf{y}\|_2^2\} \quad \Leftrightarrow \quad \min_{\mathbf{z}\in\mathbb{R}^r}\{\|\mathbf{U}_r\mathbf{\Lambda}_r^{1/2}\mathbf{z} - \mathbf{f}\|_2^2 + \lambda\|\mathbf{z}\|_2^2\}, \tag{4.1}$$

and $\widehat{\mathbf{c}}_\lambda = \mathbf{U}_r\mathbf{y}_\lambda = \mathbf{U}_r\mathbf{\Lambda}_r^{-1/2}\mathbf{z}_\lambda$. Finally, we obtain $\widehat{\phi}_\lambda = \pi(\widehat{\mathbf{c}}_\lambda) = \sum_{kj}\widehat{c}_{kj}\xi_{kj}$.

In order to select the optimal λ, we can use the L-curve [19] or GCV criterion [23]. The L-curve criterion plots the following parametrized curve in log-log scale:

$$\begin{aligned} l(\lambda) = (x(\lambda), y(\lambda)) :&= \left(\log(\|T\widehat{\phi}_\lambda - \mathbf{f}\|_2), \log(\|\phi_\lambda\|_{H_{\overline{G}}})\right) \\ &= \left(\log(\|\mathbf{\Sigma}_D\widehat{\mathbf{c}}_\lambda - \mathbf{f}\|_2), \log((\widehat{\mathbf{c}}_\lambda^\top\mathbf{\Sigma}_D\widehat{\mathbf{c}}_\lambda)^{\frac{1}{2}})\right), \end{aligned} \tag{4.2}$$

and the corner of $l(\lambda)$ corresponds to a good estimate. In practical computation, we restrict λ in the spectral range of $\mathbf{\Sigma}_D$, and compute

$$\lambda^* = \arg\max_{\lambda_r\leqslant\lambda\leqslant\lambda_1}\kappa(\lambda) := \frac{x'y'' - y'x''}{(x'^2 + y'^2)^{3/2}} \tag{4.3}$$

as the optimal λ by maximizing the signed curvature of the L-curve. For the GCV criterion, by noting that

$$\mathbf{f} - \mathbf{\Sigma}_D\widehat{\mathbf{c}}_\lambda = (\mathbf{I}_{n_0J} - \mathbf{\Sigma}_D\mathbf{\Sigma}_{D,\lambda})\mathbf{f},$$

where $\mathbf{\Sigma}_{D,\lambda} := (\mathbf{\Sigma}_D^2 + n_0J\lambda\mathbf{\Sigma}_D)^\dagger\mathbf{\Sigma}_D$, we have the following GCV function:

$$\text{GCV}(\lambda) = \frac{\|(\mathbf{I}_{n_0J} - \mathbf{\Sigma}_D\mathbf{\Sigma}_{D,\lambda})\mathbf{f}\|_2^2}{(\text{trace}(\mathbf{I}_{n_0J} - \mathbf{\Sigma}_D\mathbf{\Sigma}_{D,\lambda}))^2} = \frac{\left(\sum_{i=1}^r\left(\frac{n_0J\lambda\mathbf{u}_i^\top\mathbf{f}}{\lambda_i^2+n_0J\lambda}\right)^2 + \sum_{i=r+1}^{n_0J}(\mathbf{u}_i^\top\mathbf{f})^2\right)}{\left(n_0J - r + \sum_{i=1}^r\frac{n_0J\lambda}{\lambda_i^2+n_0J\lambda}\right)^2} \tag{4.4}$$

where we have used the numerical rank r to replace $\text{rank}(\mathbf{\Sigma}_D)$. The optimal λ is estimated as the minimizer of $\text{GCV}(\lambda)$.

Input: Data $\mathcal{D} = \{(u_k(x_j), f_k(x_j)), j = 1, \ldots, J\}_{k=1}^{n_0}$

1: Compute basis functions $\{\xi_{kj}\}$, assemble matrix $\mathbf{\Sigma}_D$ and vector $\mathbf{f}$
2: Compute the eigenvalue decomposition: $\mathbf{\Sigma}_D = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$
3: Estimate the optimal λ by L-curve or GCV criterion
4: Solve (4.1) to get $\widehat{\mathbf{c}}_\lambda = (\widehat{c}_{kj})$; compute $\widehat{\phi}_\lambda = \pi(\widehat{\mathbf{c}}_\lambda) = \sum_{k,j}\widehat{c}_{kj}\xi_{kj}$

Output: Regularized estimator $\widehat{\phi}_\lambda$

Algorithm 1: Tikhonov regularization

The algorithm of Tikhonov regularization is summarized in Algorithm 1. This method needs to store the matrix $\mathbf{\Sigma}_D$, with the memory usage of $O(n_0^2J^2)$. The main computational cost is the eigen-decomposition of $\mathbf{\Sigma}_D$, which has the order $O(n_0^3J^3)$.

4.2 Iterative regularization for large datasets

For large datasets, iterative regularization methods that rely solely on matrix-vector products are more efficient. The algorithm is based on the GKB iteration introduced in Section 2.4.

In the GKB method, the bi-diagonal structure of $\mathbf{B}_l$ in (2.25) allows us to update $\mathbf{c}_l$ step by step without explicitly solving $\min_{\mathbf{y}} \|\mathbf{B}_l\mathbf{y} - \beta_1\mathbf{e}_1\|$. The updating procedure is based on using Givens QR factorization to $\mathbf{B}_l$, which is very similar to the LSQR algorithm; see [40] for the details. In practice, we first compute an approximation $\mathbf{\Sigma}_{_D}$ to replace $\mathbf{\Sigma}$ in the computation, and then apply the GKB procedure to update the orthonormal basis of the solution subspace and the coefficient vector. The iteration will be stopped if the early stopping criterion is satisfied. The algorithm is summarized in Algorithm 2.

The hybrid regularization algorithm proceeds in the same way, except that at each iteration we update λ_l and recompute the regularized solution $\mathbf{y}_{\lambda_l}$ from (2.30). Accordingly, we omit its pseudo-code here.

Input: Data $\mathcal{D} = \{(u_k(x_j), f_k(x_j)), j = 1, \ldots, J\}_{k=1}^{n_0}$

Compute basis functions $\{\xi_{kj}\}$, assemble matrix $\mathbf{\Sigma}_{_D}$ and vector $\mathbf{f}$

(Initialization)

Compute $\bar{\mathbf{f}} = P_{\mathcal{N}(\mathbf{\Sigma}_{_D})^\perp}\mathbf{f}$, $\beta_1 = \|\bar{\mathbf{f}}\|_2$, $\mathbf{p}_1 = \bar{\mathbf{f}}/\beta_1$

Compute $\alpha_1 = \|\mathbf{p}_1\|_{\mathbf{\Sigma}_{_D}}$, $\mathbf{q}_1 = \mathbf{p}_1/\alpha_1$

Set $\mathbf{c}_0 = \mathbf{0}$, $\mathbf{w}_1 = \mathbf{q}_1$, $\bar{\varphi}_1 = \beta_1$, $\bar{\rho}_1 = \alpha_1$

for $i = 1, 2, \ldots, l_{max}$ **do**

  (GKB iteration)

  $\mathbf{r} = \mathbf{\Sigma}_{_D}\mathbf{q}_i - \alpha_i\mathbf{p}_i$, $\beta_{i+1} = \|\mathbf{r}\|_2$, $\mathbf{p}_{i+1} = \mathbf{r}/\beta_{i+1}$

  $\mathbf{s} = \mathbf{p}_{i+1} - \beta_{i+1}\mathbf{q}_i$, $\alpha_{i+1} = \|\mathbf{s}\|_{\mathbf{\Sigma}_{_D}}$, $\mathbf{q}_{i+1} = \mathbf{s}/\alpha_{i+1}$

  (Apply Givens QR factorization to $\mathbf{B}_i$)

  $\rho_i = (\bar{\rho}_i^2 + \beta_{i+1}^2)^{1/2}$

  $\bar{c}_i = \bar{\rho}_i/\rho_i$, $\bar{s}_i = \beta_{i+1}/\rho_i$

  $\theta_{i+1} = \bar{s}_i\alpha_{i+1}$, $\bar{\rho}_{i+1} = -\bar{c}_i\alpha_{i+1}$

  $\varphi_i = \bar{c}_i\bar{\varphi}_i$, $\bar{\varphi}_{i+1} = \bar{s}_i\bar{\varphi}_i$

  (Update the coefficient vector)

  $\mathbf{c}_i = \mathbf{c}_{i-1} + (\varphi_i/\rho_i)\mathbf{w}_i$, $\mathbf{w}_{i+1} = \mathbf{q}_{i+1} - (\theta_{i+1}/\rho_i)\mathbf{w}_i$

if *Early stopping criterion* is satisfied **then**

  Terminate at the estimated iteration l_*, let $\hat{\mathbf{c}} = \mathbf{c}_{l_*} = (\hat{c}_{kj})$

  Compute $\hat{\phi} = \sum \hat{c}_{kj}\xi_{kj}$

Output: Regularized estimator $\hat{\phi}$

Algorithm 2: Iterative regularization by GKB

At the initial iteration of both methods, we compute $P_{\mathcal{N}(\mathbf{\Sigma}_{_D})^\perp}\mathbf{f}$. If $\mathbf{\Sigma}_{_D}$ has full-rank or $\mathbf{f} \in \mathcal{R}(\mathbf{\Sigma}_{_D})$, then $P_{\mathcal{N}(\mathbf{\Sigma}_{_D})^\perp}\mathbf{f} = \mathbf{f}$. Otherwise, noting that $P_{\mathcal{N}(\mathbf{\Sigma}_{_D})^\perp}\mathbf{f} = \mathbf{\Sigma}_{_D}^\dagger\mathbf{\Sigma}_{_D}\mathbf{f}$, we approximate this projection by iteratively solving the minimal 2-norm least squares problem $\min_{\mathbf{v}\in\mathbb{R}^{n_0 J}} \|\mathbf{\Sigma}_{_D}\mathbf{v} - \mathbf{\Sigma}_{_D}\mathbf{f}\|_2$. This approximation does not require high accuracy, as the presence of noise limits the achievable final precision of the regularized estimator. In practice, it is carried out efficiently via the LSQR algorithm [40].

The iterative method requires $O(n_0^2 J^2)$ storage, matching the storage requirements of the

direct method of Tikhonov regularization. Each iteration is dominated by the matrix-vector product with large $\mathbf{\Sigma}_D$, costing $O(n_0^2 J^2)$ operations. Thus, over $l_{\max}$ iterations, the total computational complexity is $O\big(n_0^2 J^2\, l_{\max}\big)$. The hybrid method also incurs a total cost of $O\big(n_0^2 J^2\, l_{\max}\big)$, as the additional cost of $O(l^3)$ from the SVD of $\mathbf{B}_l$ in WGCV at each iteration is negligible compared to the dominant $O(n_0^2 J^2)$ term, placing its complexity between that of the iterative method and the direct method.

5 Numerical experiments

We present numerical results for three examples of learning kernels in operators, including integral operators, nonlocal operators, and aggregation operators in mean-field equations, as detailed in Examples 1.1–1.3. All experiments were conducted in MATLAB R2023b using double precision. The codes are available at https://github.com/Machealb/Automate-kernel.

Numerical settings. The input data $\{u_k\}$ are described in Examples 1.1–1.3 with $n_0 = 30$ and $\sigma_n = n^{-2}$. They lead to ill-conditioned and rank-deficient regression matrices with eigenvalues decaying near polynomially. We use a uniform mesh with mesh size $\Delta x = 0.005$. We use the Gaussian quadrature integrator for the integral in the operators to generate data, and use the Riemann sum to approximate it when computing the estimators. Unless otherwise specified, for all the examples we set the noise-to-signal ratio (nsr) to be $nsr = 0.1$, which corresponds to a noise with standard deviation of around $\sigma = 0.01$. Here the signal strength is the average L^2_ν-norm of the output $\{R_\phi[u_k](x_j)\}_{k,j}$.

The true kernels ϕ for the three examples are

$$\phi_1(s) = \sin(2\pi s),\ \phi_2(s) = \sin(4\pi s)\mathbf{1}_{[0,0.8]}(s),\ \phi_3(s) = -2\sin^3(6\pi s),$$

respectively, and they are plotted in Figure 2. Note that the kernel ϕ_2 of Example 1.2 has a jump discontinuity. As observed in [36], estimator accuracy improves when the smoothness of data matches that of the true kernel. Accordingly, we generate discontinuous data for Example 1.2 by multiplying each smooth u_k in Example 1.1 by the indicator of $[-0.5, 0.8]$, i.e., $u_k(y) \mapsto u_k(y)\mathbf{1}_{[-0.5,0.8]}(y)$. Furthermore, these true kernels are close to the identifiable spaces $H = \mathcal{N}(\mathcal{L}_{\overline{G}})^\perp$ for each example, making accurate estimation possible.

For each regularized estimator, we evaluate the relative L^2_μ-error with respect to the true solution, where μ is the Lebesgue measure. When reporting the statistics of the estimators (such as their means and box plots), we perform 50 independent simulations for each test.

Other regularization norms. We benchmark our $H_{\overline{G}}$-norm against two baseline norms for regularization: a Gaussian kernel norm H_K and the L^2_ρ-norm. The H_K-norm is the norm of the RKHS with the widely-used Gaussian kernel $K(s,s') = \exp(|s-s'|^2/(2\sigma_0^2))$, where the hyperparameter is $\sigma_0 = 0.1$ after fine-tuning. For both RKHS norms, we use their automatic basis functions to get an $n_0 J \times n_0 J$ linear system, and apply the Tikhonov and iterative regularization methods to compute the estimators. For the L^2_ρ-norm regularization, following Section 3, we compute the coefficients of $\hat{\phi}(s) = \sum_{l=1}^{n_s} \hat{c}_l \mathbf{1}_{I_l}(s)$ by solving the regularized least squares problem $\arg\min_{\mathbf{c}\in\mathbb{R}^{n_s}} \frac{1}{n_0 J}\|\mathbf{A}\mathbf{c}-\mathbf{f}\|_2^2 + \lambda\|\mathbf{c}\|_{\mathbf{B}}^2$, where $\mathbf{A} = \mathbf{g}\Delta s \in \mathbb{R}^{n_0 J \times n_s}$ and $\|\mathbf{c}\|_{\mathbf{B}}^2 = \mathbf{c}^\top \mathbf{B}\mathbf{c}$ with $\mathbf{B} = \mathrm{diag}(\boldsymbol{\rho}_D)$. Using the transformation $\tilde{\mathbf{c}} = \mathbf{B}^{\frac{1}{2}}\mathbf{c}$ and $\tilde{\mathbf{A}} = \mathbf{B}^{-\frac{1}{2}}\mathbf{A}$, we only need to deal with the 2-norm regularizer $\|\tilde{\mathbf{c}}\|_2$ in the Tikhonov or iterative regularization.

Accuracy of the estimators. We first compare the accuracy of the estimators computed using the three regularization norms, each with the four regularization methods: Tikhonov reg-

ularization with λ selected by the L-curve and GCV criteria, iterative regularization with early stopping determined by the L-curve, and hybrid regularization with λ_l updated by WGCV.

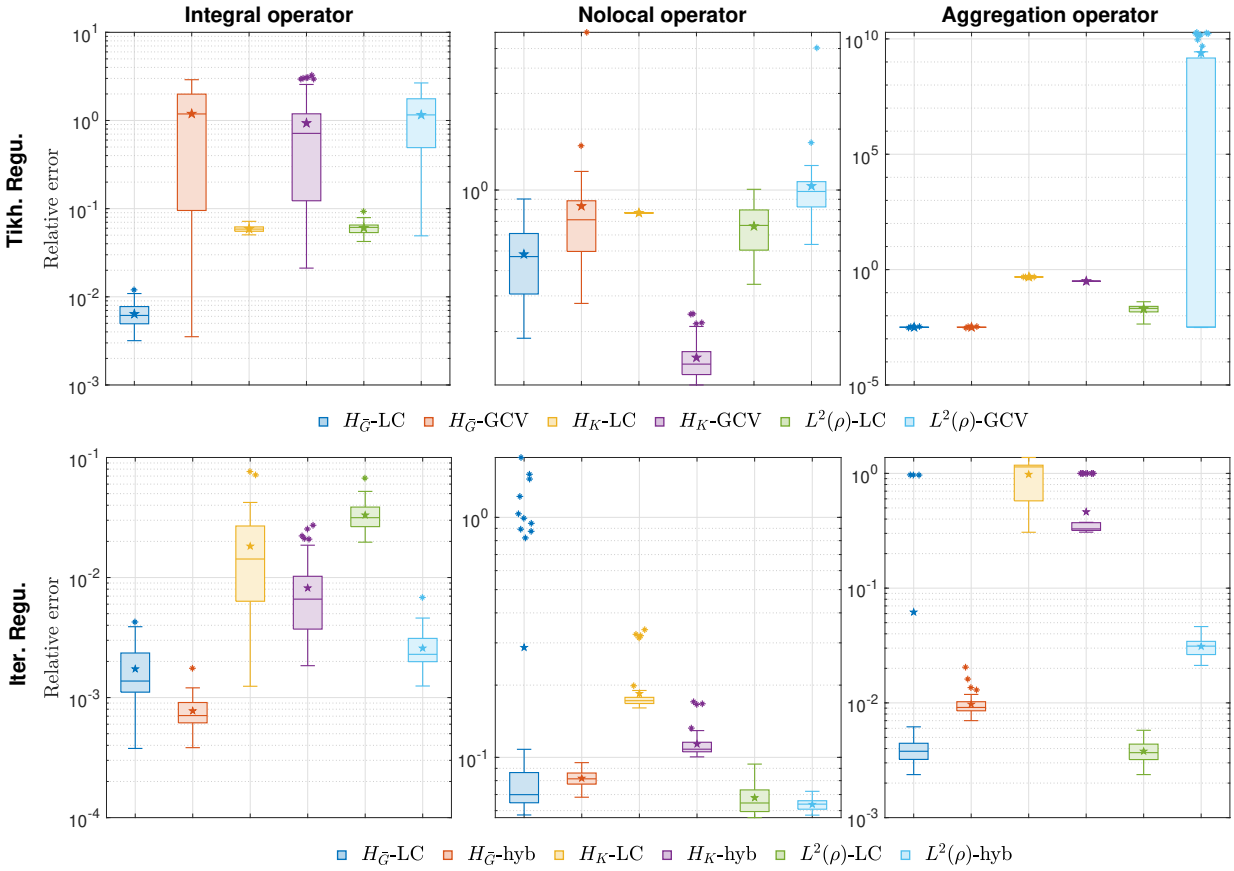

Figure 1: Relative errors of the estimators in 50 simulations.

We present the L^2_μ (μ is the Lebesgue measure) relative errors of the estimators in 50 simulations in Figure 1, where the box-plots display the median, lower and upper quartiles, outliers, and the minimum and maximum values that are not outliers. The abbreviation "LC" stands for the L-curve criterion, while "hyb" denotes the hybrid method.

The results demonstrate that the choice of regularization norm has a significant impact on the accuracy of the estimators. For all three examples, the $H_{\overline{G}}$-norm consistently yields lower relative errors, indicating that our data-adaptive RKHS regularization can better capture the structure of the underlying nonlocal inverse problems. In contrast, the Gaussian kernel norm generally yields the largest errors with large variances. While the L^2_ρ norm regularization occasionally achieves accuracy comparable to that of the $H_{\overline{G}}$-norm, such as Example 1.2 with iterative regularization methods, it is less accurate and less stable overall.

Additionally, the L-curve and GCV methods produce comparable hyperparameter selections for Tikhonov regularization (top row of Figure 1). However, the bottom row of Figure 1 illustrates that the purely iterative method can incur larger errors due to the instability of identifying the discrete L-curve's corner for early stopping. By contrast, the hybrid method offers greater stability and consistently achieves low errors across all cases.

Figure 2 displays the estimators in a typical test: only the estimators of Tikhonov with L-curve and the hybrid method are shown, since the other methods yield very similar results and

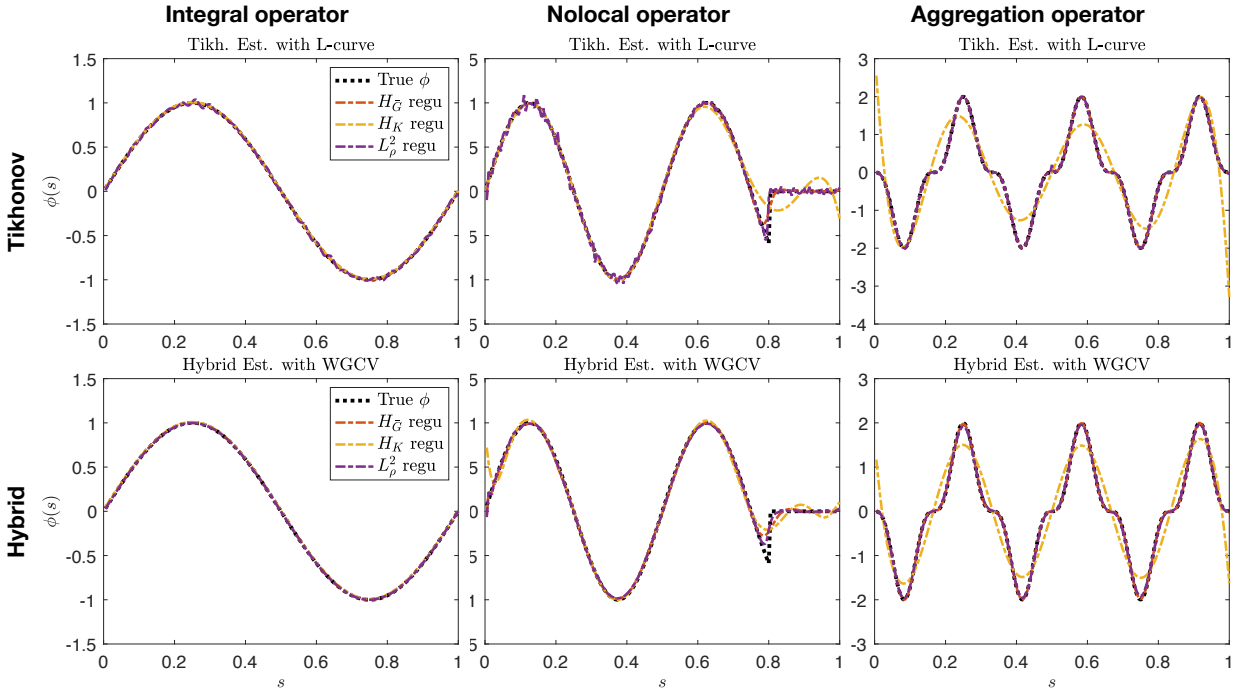

Figure 2: Typical regularized estimators by Tikhonov regularization with L-curve and hybrid regularization with WGCV using three norms: $H_{\overline{G}}$, H_K, and L^2_ρ.

are omitted for clarity. Because the noise level is low, all estimators closely track the true kernels. The $H_{\overline{G}}$-norm regularizer produces slightly more accurate estimates than the L^2_ρ and Gaussian kernel norms. In particular, for the nonlocal operator (middle column), the $H_{\overline{G}}$ regularizer better resolves the jump discontinuity than the Gaussian kernel estimator: its data-adaptive smoothness allows it to capture the discontinuity more faithfully.

Convergence as noise decreases. To compare these methods further, we examine the estimator convergence as the noise level decreases with the noise-to-signal ratio varying over $nsr \in \{1, 1/2, 1/4, 1/8, 1/16, 1/32\}$ and all other settings unchanged from the previous experiment. For each noise level, we run 50 independent simulations. In Figure 3, we report results for Tikhonov and iterative regularization using the L-curve for parameter selection, alongside the hybrid method; Tikhonov with GCV yields results similar to Tikhonov with L-curve and is omitted for clarity. To illustrate convergence behavior in the ideal scenario, we also include the relative error of the optimal iterative regularized solution (denoted by "Iter.-opt."), i.e., the solution with the minimum relative error across all iterations.

For the integral operator (left column of Figure 3), the relative error of all estimators decreases as the noise level is reduced. For all regularization methods, the estimators obtained using the $H_{\overline{G}}$-norm consistently achieve the lowest relative errors as the noise decreases.

In particular, for the optimal iterative solutions (bottom row of Figure 3), the convergence curves under the $H_{\overline{G}}$ and L^2_ρ norms are nearly identical for all the three examples, indicating that they share the same convergence rate up to a constant factor. This observation is consistent with the theoretical result presented in [37].

For the nonlocal operator (middle column of Figure 3), the Gaussian kernel regularized estimators have relatively large errors that fail to decay, due to the mismatched smoothness between

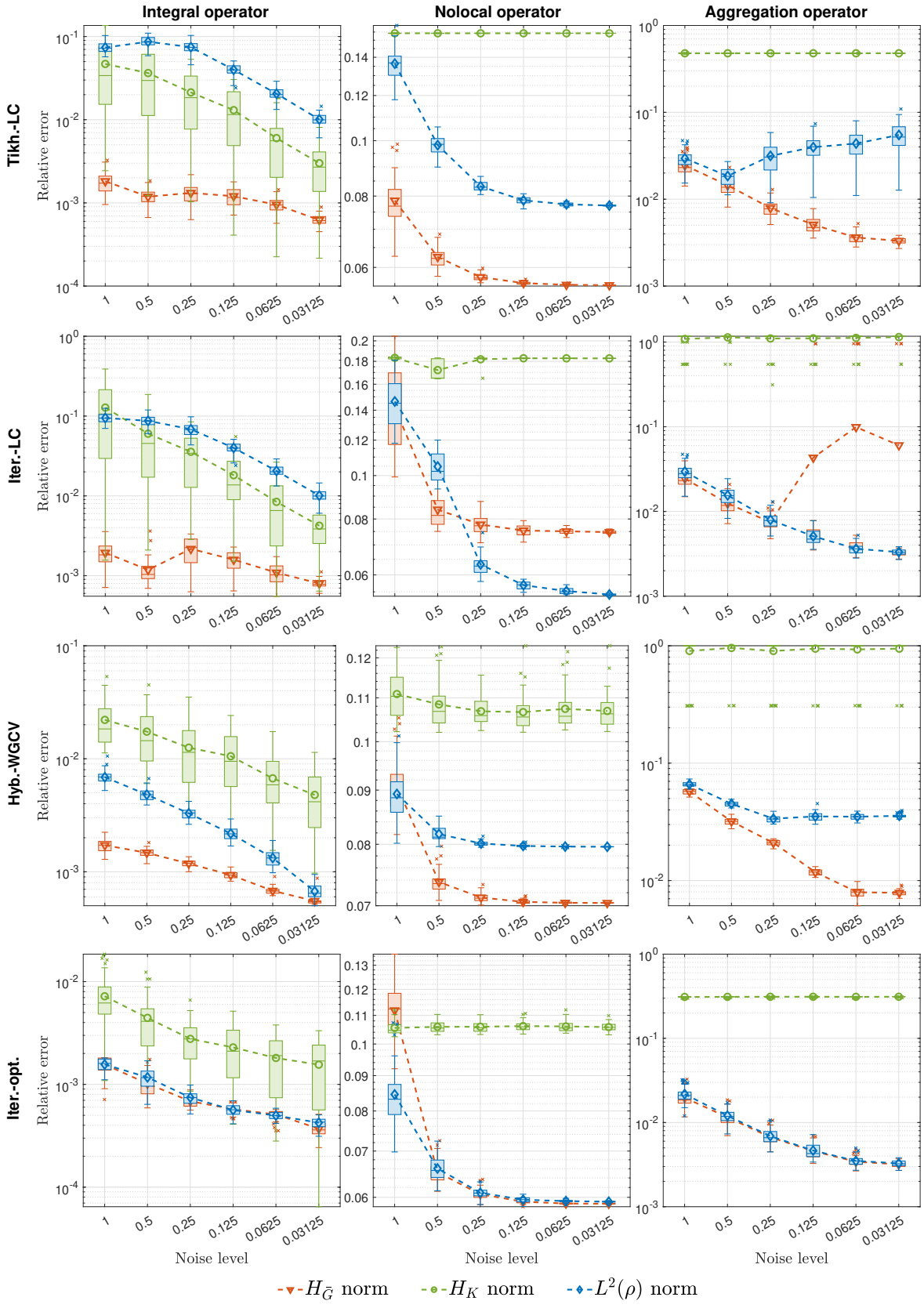

Figure 3: Convergence of the estimators as the noise decreases in 50 simulations.

the true kernel and the Gaussian kernel. In contrast, for all regularization methods, the $H_{\overline{G}}$- and L^2_ρ-norms consistently lead to estimator errors that decay with the noise, and their convergence curves become flat as the noise level approaches the numerical integration error.

The aggregation-operator case (right column of Figure 3) highlights the differences between methods. Both Tikhonov and hybrid methods yield convergent estimators under the $H_{\overline{G}}$-norm, but not under the L^2_ρ-norm. In contrast, the iterative method performs well in the L^2_ρ-norm but suffers instability in $H_{\overline{G}}$-norm due to early stopping sensitivity. All methods fail to converge under the Gaussian kernel norm, due to the relatively high frequency of the true kernel.

In summary, the $H_{\overline{G}}$-norm consistently leads to convergent estimators across nearly all regularization methods and achieves the lowest relative errors as noise decreases, outperforming both the L^2_ρ and the Gaussian kernel norms. These results confirm its effectiveness and robustness for learning convolution kernels. Moreover, the hybrid method exhibits the strongest convergence behavior overall, underscoring its ability to automatically select optimal regularization parameters and deliver accurate solutions.

Computational scalability. In this experiment, we evaluate the computational scalability of Tikhonov and iterative regularization for learning convolution kernels as the data size increases. We vary $n_0 \in \{6, 12, 18, 24, 30, 36\}$, holding all other parameters fixed as in the first experiment. We only show the results for the $H_{\overline{G}}$ regularization, as it has been proven to be the most effective in prior tests. For each value of n_0, we set the maximum number of iterations for all the three examples as $l_{\max} = 30, 30, 40, 40, 50, 50$, which is chosen to be larger than the optimal early stopping iteration. We conduct 50 independent simulations for each setting, and record the computation times on a Debian 12 desktop with 12 Intel processors.

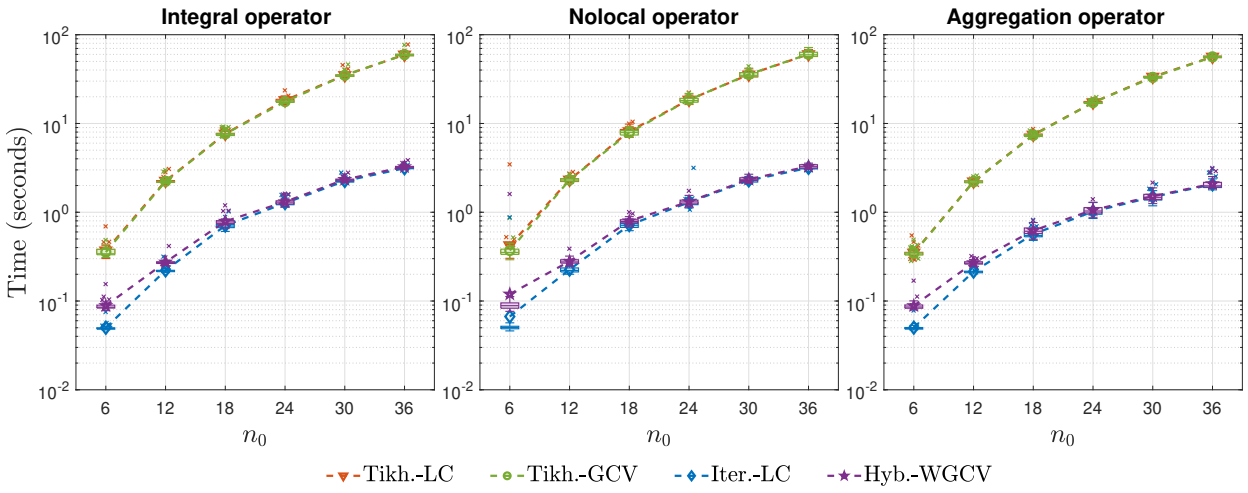

Figure 4: Running time as sample size n_0 increases for Tikhonov, iterative and hybrid regularization methods using $H_{\overline{G}}$-norm.

Figure 4 reports the computation times in these tests. The iterative regularization method is orders of magnitude faster than the Tikhonov regularization method, particularly as n_0 increases. This behavior aligns with our theoretical analysis of the computational complexity of the two approaches. Although the hybrid method incurs slightly higher runtime than pure iterative regularization, it demonstrates significantly greater stability, as evidenced by the results of the previous experiments. Therefore, for learning convolution kernels from large datasets, the hybrid

method based on iterative regularization is the most effective and reliable choice.

In summary, our numerical experiments highlight the advantages of data-adaptive RKHS regularization for learning convolution kernels. By leveraging automatically constructed basis functions, we have developed efficient and accurate iterative regularization methods that scale well with large datasets.

6 Conclusion

We have developed robust and scalable data-adaptive (DA) RKHS regularization methods for learning convolution kernels, based on an automatic reproducing kernel that is tailored to the data and the forward operator. For discrete and finite observations, the methods use a finite set of automatic basis functions sufficient to represent minimal-norm least squares, Tikhonov, and conjugate gradient estimators in the RKHS. The DA-RKHS and automatic basis functions capture the structure imposed by the forward operator and data, enabling nonparametric and mesh-free regression without the need for reproducing kernel selection, hyperparameter tuning, or predefined bases. We have developed efficient regularization algorithms, including Tikhonov methods based on matrix decompositions for small datasets and iterative methods using only matrix-vector products for large datasets. Numerical experiments on integral, nonlocal and aggregation operators demonstrate that the proposed methods outperform the ridge regression and Gaussian process regression, highlighting their effectiveness, robustness, and scalability.

A Proofs

Proof of Theorem 2.4. (a). It is clear that $\overline{G}$ is symmetric. First, we show that $\overline{G}$ is square-integrable. Since $g[u_k] \in C(\mathcal{X} \times \mathcal{S})$, we have

$$G(s,s') := \int_{\mathcal{X}} \frac{1}{n_0} \sum_{k=1}^{n_0} g[u_k](x,s) g[u_k](x,s') \, \nu(dx) \leqslant C_g \dot{\rho}(s). \tag{A.1}$$

Then, by symmetry, we obtain that $G(s,s') \leqslant C_g \min\{\dot{\rho}(s), \dot{\rho}(s')\}$ for any $s, s' \in \mathcal{S}$. Then,

$$\int_{\mathcal{S}} \int_{\mathcal{S}} \overline{G}(s,s')^2 \rho(ds) \rho(ds') = \int_{\mathcal{S}} \int_{\mathcal{S}} \frac{G(s,s')^2}{\dot{\rho}(s)\dot{\rho}(s')} ds ds' \leqslant C_g^2 |\mathrm{supp}(\rho)|^2.$$

Then, $\mathcal{L}_{\overline{G}}$ is a compact self-adjoint operator. It is positive since

$$\langle \mathcal{L}_{\overline{G}} \phi, \phi \rangle_{L^2_\rho} = \int_{\mathcal{S}} \int_{\mathcal{S}} \phi(s) \phi(s') G(s,s') ds ds' = \frac{1}{n_0} \sum_{k=1}^{n_0} \|R_\phi[u_k]\|^2 \geqslant 0$$

for any $\phi \in L^2_\rho$.

(b). By its definition in (2.1), the loss function $\mathcal{E}_{\mathcal{D}}$ can be written as (2.5). Note that $\langle \phi^D, \phi \rangle_{L^2_\rho} = \frac{1}{n_0} \sum_{k=1}^{n_0} \langle R_\phi[u_k], R_\phi[u_k] + \epsilon \rangle_{\mathbb{Y}}$ for any $\phi \in L^2_\rho$, thus, we can write ϕ^D as $\phi^D = \mathcal{L}_{\overline{G}} \phi_* + \eta$, where ϕ_* is the true kernel and $\eta \sim \mathcal{N}(0, \sigma_\epsilon^2 \mathcal{L}_{\overline{G}})$. In particular, when the data is noiseless, we have $\phi^D = \mathcal{L}_{\overline{G}} \phi_*$. Thus, the loss function $\mathcal{E}_{\mathcal{D}}$ has a unique minimizer $\widehat{\phi} = \mathcal{L}_{\overline{G}}{}^{-1} \phi^D = P_H(\phi_*)$ in $H := \overline{\mathrm{span}\{\psi_i\}_{i:\lambda_i > 0}}$.

(c). The fact that $H_{\overline{G}} = \mathcal{L}_{\overline{G}}{}^{1/2}(L^2_\rho)$ is a standard characterization of the RKHS, see, e.g., [3,13,36]. Also, for any $\phi = \sum_i c_i \psi_i \in H_{\overline{G}}$ and $\psi = \sum_i d_i \psi_i \in L^2_\rho$, using the fact that $\langle \psi_i, \psi_j \rangle_{H_{\overline{G}}} = \delta_{ij} \lambda_i^{-1}$, we have

$$\langle \phi, \mathcal{L}_{\overline{G}} \psi \rangle_{H_{\overline{G}}} = \sum_i \lambda_i^{-1} c_i d_i \lambda_i = \sum_i c_i d_i = \langle \phi, \psi \rangle_{L^2_\rho}.$$

Lastly, it follows from the definition of H that $H = \overline{H_{\overline{G}}}$. ■

Proof of Theorem 2.9. (a) First we prove that under the canonical basis of $\mathbb{R}^{n_0 J}$, it hold that $\widetilde{T}^* \mathbf{y} = \mathbf{y}$ for any $\mathbf{y} \in \mathcal{N}(\mathbf{\Sigma})^\perp$. Since

$$\langle \widetilde{T}\mathbf{x}, \mathbf{y}\rangle_2 = \langle \mathbf{x}, \widetilde{T}^*\mathbf{y}\rangle_{\mathbf{\Sigma}} \;\Leftrightarrow\; \mathbf{x}^\top \mathbf{\Sigma}(\widetilde{T}^*\mathbf{y} - \mathbf{y}) = 0, \quad \forall\, \mathbf{x} \in \mathcal{N}(\mathbf{\Sigma})^\perp,$$

we have $\mathbf{\Sigma}(\widetilde{T}^* y - y) = \mathbf{0}$. Using $\widetilde{T}^*\mathbf{y} - \mathbf{y} \in \mathcal{N}(\mathbf{\Sigma})^\perp$, we obtain $\widetilde{T}^*\mathbf{y} - \mathbf{y} = \mathbf{0}$, which is the desired result. Using the basic property of the GKB process, $\{\mathbf{q}_i\}_{i=1}^m$ and $\{\mathbf{p}_i\}_{i=1}^m$ are the $\mathbf{\Sigma}$-orthonormal and 2-orthonormal bases of the Krylov subspaces

$$\mathcal{K}_l(\widetilde{T}^*\widetilde{T}, \widetilde{T}^* P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}) = \mathrm{span}\{(\widetilde{T}^*\widetilde{T})^i \widetilde{T}^* P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}\}_{i=0}^{l-1} = \mathrm{span}\{\mathbf{\Sigma}^i P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}\}_{i=0}^{l-1},$$
$$\mathcal{K}_l(\widetilde{T}\widetilde{T}^*, P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}) = \mathrm{span}\{(\widetilde{T}\widetilde{T}^*)^i P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}\}_{i=0}^{l-1} = \mathrm{span}\{\mathbf{\Sigma}^i P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}\}_{i=0}^{l-1},$$

respectively. The last relation is obvious since $\mathbf{\Sigma}^i P_{\mathcal{N}(\mathbf{\Sigma})^\perp} = \mathbf{\Sigma}^{i+1}\mathbf{\Sigma}^\dagger$.

(b) Since $\{\mathbf{p}_i\}$ and $\{\mathbf{q}_i\}$ are 2-orthonormal and $\mathbf{\Sigma}$-orthonormal bases, the maximum GKB iteration must not exceed the dimension $\mathcal{N}(\mathbf{\Sigma})^\perp$, which is $\mathrm{rank}(\mathbf{\Sigma})$, that is, $l_t \leqslant \mathrm{rank}(\mathbf{\Sigma})$. Using Theorem 2.8 and that $\mathcal{H}_m = \pi(\mathcal{K}_l)$, $\mathcal{K}_l \subset \mathcal{N}(\mathbf{\Sigma})^\perp$, and $\pi|_{\mathcal{N}(\mathbf{\Sigma})^\perp} \to H_{\overline{G}}$ is injective, there is a one-to-one correspondence between the CG for (2.19) and the CG for $\min_{\mathbf{c}\in\mathcal{N}(\mathbf{\Sigma})^\perp} \|\widetilde{T}\mathbf{c} - P_{\mathcal{N}(\mathbf{\Sigma})^\perp}\mathbf{f}\|_2$. Therefore, the CG for T and $\widetilde{T}$ terminate at the same step, the basic property of CG implies that $\widetilde{\phi} = \phi_{l_t} = \pi(\mathbf{c}_{l_t})$.

(c) Using $T \circ \pi = \mathbf{\Sigma}$, we have $T\phi_l - \mathbf{f} = T \circ \pi(\mathbf{c}_l) - \mathbf{f} = \mathbf{\Sigma}\mathbf{c}_l - \mathbf{f}$. Using (2.8) we get $\|\phi_l\|_{H_{\overline{G}}} = \|\mathbf{c}_l\|_{\mathbf{\Sigma}}$. The basic property of CG for T states that with a zero initial solution, the residual norm monotonically decreases and the solution norm increases. This is the last assertion. ■

Acknowledgments. The work of FL is partially funded by the NSF DMS2238486 and AFOSR FA9550-21-1-0317. The authors would like to thank Jinchao Feng for helpful discussions.

References

[1] Anima Anandkumar, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Nikola Kovachki, Zongyi Li, Burigede Liu, and Andrew Stuart. Neural operator: Graph kernel network for partial differential equations. In *ICLR 2020 workshop on integration of deep neural models and differential equations*, 2020.

[2] David Applebaum. *Lévy processes and stochastic calculus*. Cambridge university press, 2009.

[3] Nachman Aronszajn. Theory of reproducing kernels. *Transactions of the American mathematical society*, 68(3):337–404, 1950.

[4] David A Benson, Stephen W Wheatcraft, and Mark M Meerschaert. Application of a fractional advection-dispersion equation. *Water resources research*, 36(6):1403–1412, 2000.

[5] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. Image denoising methods. a new nonlocal principle. *SIAM review*, 52(1):113–147, 2010.

[6] José A Carrillo, Katy Craig, and Yao Yao. Aggregation-diffusion equations: dynamics, asymptotics, and singular limits. In *Active Particles, Volume 2*, pages 65–108. Springer, 2019.

[7] Noe Angelo Caruso and Paolo Novati. Convergence analysis of LSQR for compact operator equations. *Linear Algebra and its Applications*, 583:146–164, 2019.

[8] Neil K Chada, Quanjun Lang, Fei Lu, and Xiong Wang. A data-adaptive RKHS prior for Bayesian learning of kernels in operators. *Journal of Machine Learning Research*, 25(317):1–37, 2024.

[9] Yifan Chen, Bamdad Hosseini, Houman Owhadi, and Andrew M Stuart. Solving and learning nonlinear PDEs with gaussian processes. *Journal of Computational Physics*, 447:110668, 2021.

[10] Zhen-Qing Chen and Xicheng Zhang. Heat kernels for non-symmetric non-local operators. *Recent developments in nonlocal theory*, pages 24–51, 2018.

[11] Julianne Chung, James G Nagy, and Dianne P O'Leary. A weighted-GCV method for Lanczos-hybrid regularization. *Electr. Trans. Numer. Anal.*, 28(29):149–167, 2008.

[12] Felipe Cucker and Steve Smale. On the mathematical foundations of learning. *Bulletin of the American mathematical society*, 39(1):1–49, 2002.

[13] Felipe Cucker and Ding Xuan Zhou. *Learning theory: an approximation theory viewpoint*, volume 24. Cambridge University Press, Cambridge, 2007.

[14] Matthieu Darcy, Boumediene Hamzi, Giulia Livieri, Houman Owhadi, and Peyman Tavallali. One-shot learning of stochastic differential equations with data adapted kernels. *Physica D: Nonlinear Phenomena*, 444:133583, 2023.

[15] Matthieu Darcy, Boumediene Hamzi, Jouni Susiluoto, Amy Braverman, and Houman Owhadi. Learning dynamical systems from data: a simple cross-validation perspective, part ii: nonparametric kernel flows. *Physica D: Nonlinear Phenomena*, 476:134641, 2025.

[16] Marta D'Elia, Qiang Du, Christian Glusa, Max Gunzburger, Xiaochuan Tian, and Zhi Zhou. Numerical methods for nonlocal and fractional models. *Acta Numerica*, 29:1–124, 2020.

[17] Qiang Du, Max Gunzburger, Richard B Lehoucq, and Kun Zhou. Analysis and approximation of nonlocal diffusion problems with volume constraints. *SIAM review*, 54(4):667–696, 2012.

[18] Jinqiao Duan. *An introduction to stochastic dynamics*, volume 51. Cambridge University Press, 2015.

[19] Heinz W Engl and Wilhelm Grever. Using the l–curve for determining optimal regularization parameters. *Numerische Mathematik*, 69(1):25–31, 1994.

[20] Heinz Werner Engl, Martin Hanke, and Andreas Neubauer. *Regularization of inverse problems*, volume 375. Springer Science & Business Media, 1996.

[21] Jinchao Feng, Charles Kulick, Yunxiang Ren, and Sui Tang. Learning particle swarming models from data with gaussian processes. *Mathematics of Computation*, 93(349):2391–2437, 2024.

[22] Guy Gilboa and Stanley Osher. Nonlocal operators with applications to image processing. *Multiscale Modeling & Simulation*, 7(3):1005–1028, 2009.

[23] Gene H Golub, Michael Heath, and Grace Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

[24] László Györfi, Michael Kohler, Adam Krzyzak, and Harro Walk. *A distribution-free theory of nonparametric regression*. Springer Science & Business Media, 2006.

[25] Per Christian Hansen. *Discrete inverse problems: insight and algorithms*. SIAM, 2010.

[26] John Harlim, Daniel Sanz-Alonso, and Ruiyi Yang. Kernel methods for bayesian elliptic inverse problems on manifolds. *SIAM/ASA Journal on Uncertainty Quantification*, 8(4):1414–1445, 2020.

[27] Amit Katiyar, Shivam Agrawal, Hisanao Ouchi, Pablo Seleson, John T Foster, and Mukul M Sharma. A general peridynamics model for multiphase transport of non-newtonian compressible fluids in porous media. *Journal of Computational Physics*, 402:109075, 2020.

[28] Misha E Kilmer and Dianne P O'Leary. Choosing regularization parameters in iterative methods for ill-posed problems. *SIAM J. Matrix Anal. Appl.*, 22(4):1204–1221, 2001.

[29] Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces with applications to pdes. *Journal of Machine Learning Research*, 24(89):1–97, 2023.

[30] Quanjun Lang and Fei Lu. Identifiability of interaction kernels in mean-field equations of interacting particles. *Foundations of Data Science*, 5(4):480–502, 2023.

[31] Quanjun Lang and Fei Lu. Small noise analysis for Tikhonov and RKHS regularizations. *arXiv preprint arXiv:2305.11055*, 2023.

[32] Haibo Li. A preconditioned krylov subspace method for linear inverse problems with general-form tikhonov regularization. *SIAM Journal on Scientific Computing*, 46(4):A2607–A2633, 2024.

[33] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

[34] Yifei Lou, Xiaoqun Zhang, Stanley Osher, and Andrea Bertozzi. Image recovery via nonlocal operators. *Journal of Scientific Computing*, 42(2):185–197, 2010.

[35] Fei Lu, Qingci An, and Yue Yu. Nonparametric learning of kernels in nonlocal operators. *Journal of Peridynamics and Nonlocal Modeling*, pages 1–24, 2023.

[36] Fei Lu, Quanjun Lang, and Qingci An. Data adaptive RKHS Tikhonov regularization for learning kernels in operators. *Proceedings of Mathematical and Scientific Machine Learning, PMLR 190:158-172*, 2022.

[37] Fei Lu and Miao-Jung Yvonne Ou. An adaptive RKHS regularization for the Fredholm integral equations. *Mathematical Methods in the Applied Sciences*, 2025.

[38] Lu Lu, Pengzhan Jin, and George Em Karniadakis. Deeponet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. *arXiv preprint arXiv:1910.03193*, 2019.

[39] Houman Owhadi and Gene Ryan Yoo. Kernel flows: From learning kernels from data into the abyss. *Journal of Computational Physics*, 389:22–47, 2019.

[40] Christopher C Paige and Michael A Saunders. LSQR: An algorithm for sparse linear equations and sparse least squares. *ACM Transactions on Mathematical Software (TOMS)*, 8(1):43–71, 1982.

[41] Stewart A Silling, Michael Epton, Olaf Weckner, Jifeng Xu, and Ebrahim Askari. Peridynamic states and constitutive modeling. *Journal of elasticity*, 88:151–184, 2007.

[42] Grace Wahba. Convergence rates of certain approximate solutions to fredholm integral equations of the first kind. *Journal of Approximation Theory*, 7(2):167–185, 1973.

[43] Grace Wahba. Practical approximate solutions to linear operator equations when the data are noisy. *SIAM journal on numerical analysis*, 14(4):651–667, 1977.

[44] Grace Wahba. *Spline models for observational data*. SIAM, 1990.

[45] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018.

[46] Huaiqian You, Yue Yu, Stewart Silling, and Marta D'Elia. A data-driven peridynamic continuum model for upscaling molecular dynamics. *Computer Methods in Applied Mechanics and Engineering*, 389:114400, 2022.

[47] Huaiqian You, Yue Yu, Stewart Silling, and Marta D'Elia. Nonlocal operator learning for homogenized models: From high-fidelity simulations to constitutive laws. *Journal of Peridynamics and Nonlocal Modeling*, 6(4):709–724, 2024.

[48] Ming Yuan and T Tony Cai. A reproducing kernel Hilbert space approach to functional linear regression. *The Annals of Statistics*, 38(6):3412–3444, 2010.

[49] He Zhang, John Harlim, and Xiantao Li. Estimating linear response statistics using orthogonal polynomials: An rkhs formulation. *Foundations of Data Science*, 2(4):443–485, 2020.

[50] Sichong Zhang, Xiong Wang, and Fei Lu. Minimax rate for learning kernels in operators. *arXiv preprint arXiv:2502.20368*, 2025.