Journal of Computational Physics 507 (2024) 112971

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Learning nonparametric ordinary differential equations from noisy data

Kamel Lahouel [b], Michael Wells [a], Victor Rielly [a], Ethan Lew [c], David Lovitz [a], Bruno M. Jedynak [a,*]

[a] Dept. of Math & Stat, Portland State University, 1855 SW Broadway, Portland, OR 97201, United States of America
[b] TGen, 445 N. Fifth Street, Phoenix, AZ 85004, United States of America
[c] Galois Inc., 421 SW 6th Avenue, Suite 300, Portland, OR 97204, United States of America

ARTICLE INFO

Keywords:
Nonlinear dynamical systems
System identification
Kernel methods
Penalty method
Amyloid accumulation

ABSTRACT

Learning nonparametric systems of Ordinary Differential Equations (ODEs) $\dot{x} = f(t, x)$ from noisy data is an emerging machine learning topic. We use the well-developed theory of Reproducing Kernel Hilbert Spaces (RKHS) to define candidates for f for which the solution of the ODE exists and is unique. Learning f consists of solving a constrained optimization problem in an RKHS. We propose a penalty method that iteratively uses the Representer theorem and Euler approximations to provide a numerical solution. We prove a generalization bound for the L^2 distance between x and its estimator. Experiments are provided for the FitzHugh–Nagumo oscillator, the Lorenz system, and for predicting the Amyloid level in the cortex of aging subjects. In all cases, we show competitive results compared with the state-of-the-art.

1. Introduction

1.1. Description of the problem and related works

Fitting a system of nonparametric ordinary differential equations (ODEs) $\dot{x} = f(t, x)$ to longitudinal data could lead to scientific breakthroughs in disciplines where ODEs or dynamical systems have been used for a long time, including physics, chemistry, and biology, see [1]. By nonparametric, we mean that there is no need to specify the functional form of the vector-field f using a pre-defined finite dimensional parameter. Instead, this force field belongs to a functional space and the number of parameters that characterize this vector field depends on the amount of data available. This provides a great advantage in situations where the form of the vector field is unknown but data is available for learning. The functional spaces considered are Reproducing Kernel Hilbert Spaces (RKHS) [2], allowing for efficient optimization among other desirable properties.

A particular difficulty arises when the data is sparse and noisy. This is often the case for longitudinal healthcare data obtained during hospital visits. These visits provide measurements that are sparse in time, with a high level of individual variability. The work

* Corresponding author.
E-mail addresses: klahouel@tgen.org (K. Lahouel), mlwells@pdx.edu (M. Wells), victor23@pdx.edu (V. Rielly), elew@galois.com (E. Lew), lovitz@pdx.edu (D. Lovitz), bruno.jedynak@pdx.edu (B.M. Jedynak).

https://doi.org/10.1016/j.jcp.2024.112971
Received 20 February 2023; Received in revised form 6 March 2024; Accepted 24 March 2024

Available online 29 March 2024

presented in this paper has been motivated in part by the need to model the accumulation of the Amyloid protein in the brain of aging subjects. Understanding how amyloid contributes to the manifestation of Alzheimer's is a crucial task. The algorithm discussed here will (we hope) shed more light on the development of this devastating disease.

Fitting data to nonparametric ODEs is an inverse problem. It requires making assumptions on the initial state of the solution and on the vector field. Furthermore, one needs to make assumptions about the noise model and provide a tractable optimization algorithm.

We now provide a short bibliographic survey. Further references can be found in the cited papers. First, note that if the time derivative ($\dot{x}$) was observed, then fitting ODEs to noisy data would reduce to solving a regression problem. This remark has led to the methods known as "gradient matching" and to the earliest success in fitting ODEs to data, see e.g. [3,4]. It consists in estimating the gradient from the data, then performing nonparametric regression to fit the vector field f and eventually, iterating, see [5]. These methods become inefficient when the data is sparse and/or noisy.

Another approach consists in modeling f with polynomials [6]. Alternatively, one could model f using the units of a Deep Neural Network, see [7,8]. These methods integrate the solution along the vector field from guessed initial conditions and compare the resulting trajectories with the observations. Optimization is used iteratively to refine the estimation of f and the initial conditions. Stochastic gradient descent and backpropagation is used in the latter case. Another modeling approach is to assume that f belongs to an RKHS. This idea, also known under the name of kernel method, could be traced back to [9]. It was successfully applied to fluid mechanics in [10]. This is the conceptual approach pursued here. We believe that this approach is well-motivated since there is a tight connection between the regularity (smoothness) properties of a kernel and the regularity properties of f. Specifically, one can choose an RKHS of vector-valued functions for which one is guaranteed the existence and uniqueness of the corresponding initial value problem. This is a necessary step in proving that more data would result in more accurate predictions. Another advantage of kernel methods is that there is no need to choose a dictionary of functions as in [4]. Instead, one selects a kernel, which, our experiments suggest, is easier. In [11], the authors assume that each coordinate of the trajectory belongs to a real-valued RKHS where the functions' input is time. In their approach, they first retrieve the full trajectory solving a kernel ridge regression problem. Next, they solve for the vector field given the full trajectory, assuming that each coordinate of the vector field can be written as a sum of a linear combination of functions, which are defined on each coordinate of the trajectory. Our framework allows for linear combinations of pairwise products of such functions, as well. The functions characterizing such a vector field are assumed to be in a real-valued RKHS taking a single coordinate as input. In our approach, we make an assumption on the vector field. This soft constraint translates to a soft constraint on the set of trajectories, without imposing additional constraints on the trajectory itself. As a result, we solve one optimization problem as opposed to the two-step approach in [11]. Moreover, we allow for higher-order interaction terms compared to the pairwise single coordinates interaction assumed in the mentioned work. In [12], the authors use a Gaussian process (GP) for the vector field. This is the Bayesian counterpart of the frequentist RKHS modeling, see [13] for a review of the similarities and differences between RKHSs and GPs. Comparisons between a collection of algorithms representative of the state of the art and the proposed algorithm is provided in the experiment section.

For the purpose of providing a visual and easy to understand illustration of the results generated by the algorithms presented in this paper, please see Fig. 1. The details of this experiment are provided in section 4.3.1. We see that the proposed algorithm is able to recover a noisy trajectory and extrapolate the data, contrary to a method that would use a regression model and ignore the ODE.

1.2. Main contributions

The main contributions of this paper are as follows:

1. We present an RKHS model for fitting nonparametric ODEs to observational data. Conditions for existence and uniqueness of the solutions of the corresponding initial value problem are expressed in terms of the regularity of the kernel;
2. We propose a novel algorithm for estimating nonparametric ODEs and the initial condition(s) from noisy data. This algorithm solves a constrained optimization problem using a penalty method;
3. We derive and prove a consistency result for the prediction of the state (interpolation) at unobserved times. This is, up to our knowledge, the first result for the problem of fitting nonparametric ODEs to data.
4. We provide experiments with simulated data. We compare the proposed algorithm to 6 existing methods representing state of the art for various noise levels. We show that the ODE-RKHS algorithm is competitive.
5. We provide an experiment modeling the accumulation of Amyloid in the cortex of aging subjects. The data is sparse with, on average, three data points per trajectory (subject) and 179 trajectories. We show competitive performance compared to state of the art.

The rest of this paper is organized as follows: Section 2 presents some background material as well as the model and the algorithms. The consistency results are presented in Section 3 and proved in Appendix A. The experiments appear in Section 4 while Section 5 provides concluding remarks. Appendix B provides examples of kernels.

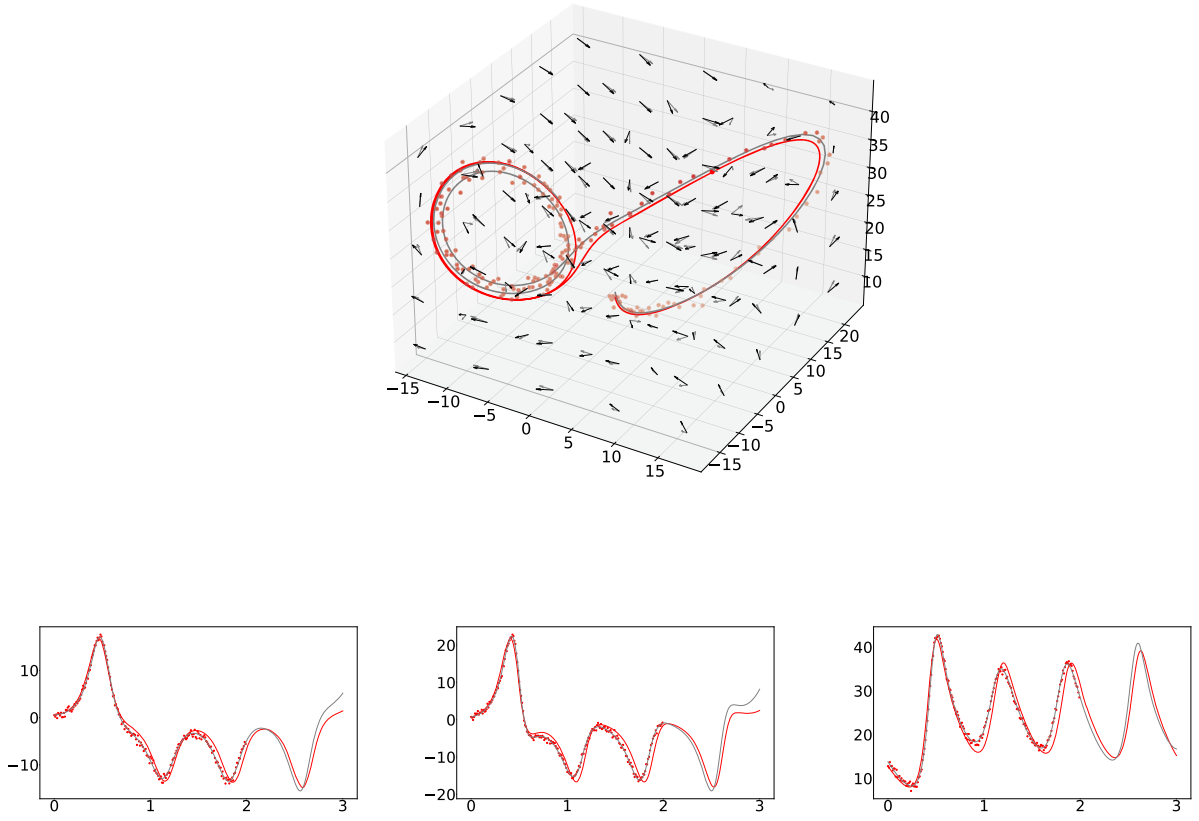


Fig. 1. (a) Predicted vector field of the Lorenz system. The Black arrows are the prediction and the grey are the true vector field. Red points are observations. The red curve is a predicted trajectory while the grey is the true trajectory. (b) is the x-dimension, (c) is the y-dimension and (d) is the z-dimension. The red points are the observations. This plot also shows a prediction beyond the last observation in the data.

2. Model and algorithm

2.1. Background on Reproducing Kernel Hilbert Spaces (RKHSs)

Basic notions and notations associated with RKHS are important for understanding the algorithms and derivations presented in this paper. We thus provide a short presentation. We limit ourselves to RKHS over the field of real numbers instead of complex numbers as this is sufficient throughout this paper. We begin with the univariate real-valued case and we continue with the vector-valued case which allows us to describe vector fields, central to this paper.

2.1.1. Real-valued RKHS

Real-valued RKHS are Hilbert spaces of real-valued functions: $\mathcal{X} \to \mathbb{R}$, where $\mathcal{X}$ is a nonempty space. The critical assumption which make them "reproducing" is that the evaluation functional is continuous. The evaluation functional at $x \in \mathcal{X}$ is a mapping from a RKHS H to $\mathbb{R}$, which associates to a function its evaluation at x, that is $f \mapsto f(x)$. Thanks to the Riesz representation theorem, evaluating a function in an RKHS is a geometric operation consisting in computing an inner product. Effectively, for any $x \in \mathcal{X}$, there is a unique vector $k_x \in H$ such that

$$f(x) = \langle f, k_x \rangle_H \tag{1}$$

where $\langle .,. \rangle_H$ is the scalar product associated with H. In what follows, we will simply notate $\langle .,. \rangle$ for this inner product. Moreover, let us define, for any $x, y \in \mathcal{X}$, the so-called kernel

$$k(x, y) = \langle k_x, k_y \rangle \tag{2}$$

and let us use this to characterize the function k_x. Evaluating k_x at y and using Riesz representation provides

$$k_x(y) = \langle k_x, k_y \rangle = \langle k_y, k_x \rangle = k(y, x) \tag{3}$$

Thus the function $k_x(.)$ is the function $k(., x)$ and for any $f \in H$,

$$f(x) = \langle f, k(.,x) \rangle \tag{4}$$

This is the reproducible property of the kernel. Replacing the function f by k_y, and using (3), we obtain that

$$k_y(x) = \left\langle k_y, k(.,x) \right\rangle = \langle k(.,y), k(.,x) \rangle = k(y,x) \tag{5}$$

2.1.2. Vector-valued RKHSs

Vector-valued RKHSs generalize the real-valued case. The construction is similar. Consider a Hilbert space of functions from $\mathcal{X}$ to $\mathbb{R}^d$. Assume, moreover, as in the real-valued case, that the evaluation functional is continuous. The Riesz representation theorem then states that for any $x \in \mathcal{X}$, and $v \in \mathbb{R}^d$, there exists a unique element in H, notated $K_{x,v}$ such that $v^T f(x) = \left\langle f, K_{x,v} \right\rangle$. The kernel of H is then the (d,d) matrix where the element (i,j) at the i^{th} row and j^{th} column is defined by

$$K_{ij}(x,y) = \left\langle K_{x,e_i}, K_{y,e_j} \right\rangle \tag{6}$$

where $(e_1, \dots, e_d)$ is the natural basis of $\mathbb{R}^d$. Let us use (6) to characterize the function $K_{x,v}$. We start with K_{y,e_j} and use the reproducing property as well as the symmetry of the inner product.

$$e_i^T K_{y,e_j}(x) = \left\langle K_{y,e_j}, K_{x,e_i} \right\rangle = \left\langle K_{x,e_i}, K_{y,e_j} \right\rangle = K_{ij}(x,y) = e_i^T K(x,y) e_j \tag{7}$$

Thus $K_{y,e_j}(.) = K(.,y)e_j$, and

$$v^T f(x) = \left\langle f, K_{x,v} \right\rangle = \langle f, K(.,x)v \rangle \tag{8}$$

which is the reproducing property for vector-valued RKHS. Applying (8) to the function $x \mapsto K(x,y)w$, for $w \in \mathbb{R}^d$ provides

$$v^T K(x,y) w = \langle K(.,y)w, K(.,x)v \rangle = \langle K(.,x)v, K(.,y)w \rangle \tag{9}$$

Lastly, a useful property of the kernel K is that $K(x,y)^T = K(y,x)$. Indeed,

$$K_{ji}(x,y) = e_j^T K(x,y) e_i = \left\langle K(.,x)e_j, K(.,y)e_i \right\rangle = \left\langle K(.,y)e_i, K(.,x)e_j \right\rangle = e_i^T K(y,x) e_j = K_{ij}(y,x) \tag{10}$$

Choosing $\mathcal{X} = \mathbb{R}^d$ allows for defining autonomous vector fields, that is functions $\mathbb{R}^d \to \mathbb{R}^d$, and choosing a suitable kernel allows for choosing Lipschitz continuous vector fields as will be discussed in Section 2.3.

2.2. Notations

The observations are characterized by multiple time series. There are n times series. The i^{th} one is of length m_i. It is characterized by m_i couples $(t_{ij}, y_{ij}(t_{ij})), i = 1, \dots, m_i$, where $t_{ij} \in [0,T]$ for some maximum predefined time T, and the observations $y_{ij}(t_{ij})$ belong to $\mathbb{R}^d$.

We aim to make predictions at new time points along a time series having one or several noisy snapshots. To this end, we explore the following nonparametric ODE model:

$$\begin{cases} \dot{x} &= f(t,x) \\ y_{ij}(t_{ij}) &= x(t_{ij}) + \epsilon_{ij} \end{cases} \tag{11}$$

where $i = 1, \dots, n$, $j = 1, \dots, m_i$. The noise ϵ_{ij} is bounded or sub-Gaussian. This model is nonparametric because f is not specified parametrically. We assume that f belongs to a RKHS of smooth functions for which the solution of the ODE exists and is unique, see Section 2.3. Background material on RKHS can be found in [14] and vector-valued RKHS are reviewed in [15]. The rest of the paper is written for the autonomous case when $f(t,x) = f(x)$ and for the simpler situation where m_i is the same for all time series and when the time points t_{ij} are the same for all the time series i.e. do not depend on i. However, we will point to the modifications for the non-autonomous setting when necessary, as well as the situation of non regular sampling.

2.3. Existence and uniqueness

It is a classical result, see [16], that the initial value problem (IVP):

$$\dot{x}(t) = f(x(t)) \text{ and } x(0) = x_0, \tag{12}$$

where $f : \mathbb{R}^d \to \mathbb{R}^d$ is Lipschitz continuous has a unique solution defined on the domain $[0, +\infty)$.

Let H be an RKHS of vector-valued functions $\mathbb{R}^d \mapsto \mathbb{R}^d$ and let K be the reproducing kernel of H. K is a (d,d) matrix-valued kernel. It is then natural to ask: what is a sufficient condition on K which ensures that all $f \in H$ are Lipschitz continuous? The following lemma provides an answer.

Lemma 1. *If $f : \mathbb{R}^d \to \mathbb{R}^d$ belongs to an RKHS with kernel K such that:*

$$d^2_{K_{ii}}(u,v) := K_{ii}(u,u) - 2K_{ii}(u,v) + K_{ii}(v,v) \leq N_K^2|u-v|^2, \forall u,v \in \mathbb{R}^d, i = 1\dots d, \quad (13)$$

for some constant N_K, then the IVP problem (12) has a unique solution defined on $[0,+\infty)$.

Proof. Notice that for every $i = 1,\dots,d$

$$|f_i(u) - f_i(v)|^2 = |\langle K(u,\cdot)e_i - K(v,\cdot)e_i, f\rangle_H|^2 \quad (14)$$

$$\leq ||K(u,\cdot)e_i - K(v,\cdot)e_i||_H^2 ||f||_H^2 \quad (15)$$

$$= d^2_{K_{ii}}(u,v)||f||_H^2 \quad (16)$$

where $e = (e_1,\dots,e_d)$ is the natural basis of $\mathbb{R}^d$. Here we have used the reproducing property of the matrix-valued kernel and the Cauchy-Schwartz inequality. □

Thus, one can choose a kernel that guarantees the existence and uniqueness of the solution of the IVP, which will lead to provable asymptotic performance. We believe that this simple result is a good motivator for the proposed modeling approach.

Let us discuss some examples of kernels satisfying Lemma 1. The simplest matrix-valued kernels are separable kernels. They are obtained by choosing a scalar kernel K_1 and a positive semi-definite matrix A. Then,

$$K(x,y) = K_1(x,y)A \quad (17)$$

The diagonal elements of K are then positive multiples of K_1. Thus, if K_1 verifies the regularity condition of Lemma 1, then so do all the separable kernels based on K_1. The scalar kernels satisfying the hypothesis of Lemma 1 contain the linear kernel, the Gaussian Kernel, the rational quadratic kernel, the sinc kernel and the Matérn kernels for $p > 3/2$. Kernels for which the functions in their corresponding RKHSs are not guaranteed to provide unique solutions to the corresponding IVP due to lack of regularity include the polynomial kernels with an order of at least two, the Laplacian kernel and the Matérn kernel for $p \leq 3/2$. Details are provided in Appendix B. The condition of Lemma 1 has a nice interpretation in the case where explicit kernels are used. Indeed, when a feature map associated with the kernel is given explicitly, the conditions of Lemma 1 are equivalent to assuming Lipschitz continuous features. The details are provided in the Appendix B.

Note on the non-autonomous case: When the vector field is time-dependent denoted by $f(t,x)$, the kernel is defined on $[0,\infty)\times\mathbb{R}^d$. It is sufficient to assume a global Lipschitz condition with respect to the second variable [16], namely: There exists a constant L_K such that for every $t \geq 0$ and $u,v \in \mathbb{R}^d$ and $i \in 1,...,d$:

$$|f_i(t,u) - f_i(t,v)| \leq L_K|u-v| \quad (18)$$

It is therefore sufficient to assume a kernel K defined on $[0,\infty)\times\mathbb{R}^d$ and satisfying the conditions of Lemma 1 as it will ensure the following inequality:

$$d^2_{K_{ii}}(t,u,t,v) \leq N_K^2|u-v|^2 \quad (19)$$

2.4. From constrained to unconstrained optimization

We first construct the optimization algorithm in the case $n = 1$. All the observations are from a single trajectory with the same initial condition. Thus, we temporarily drop the double indexing with subjects and times to simplify the notation.

Assume the observation times are $t_1 < \dots < t_m$. Consider the following constrained minimization problem:

$$\min_{x,f} \frac{1}{m}\sum_{j=1}^{m}|y_j - x(t_j)|^2 + \lambda||f - f_0||_H^2, \quad (20)$$

under the constraints

$$\begin{cases} f \in H, \text{ the RKHS with matrix-valued kernel } K, \\ x(t) = x(t_1) + \int_{t_1}^{t} f(x(s))ds, \text{ for } t_1 \leq t \leq t_m. \end{cases} \quad (21)$$

The function $f_0 \in H$ is an initial guess for f. Section 2.6 describes a gradient matching algorithm for selecting f_0. K is a kernel that satisfies Lemma 1.

Consider a regular one-dimensional grid over the interval $[t_1,t_m]$. Specifically, we choose

$$s_l = t_1 + lh \quad (22)$$

with $l = 0,\dots,k$ and we assume that h is small enough so that there are integers $k_1 = 0 < k_2 < \dots < k_m$, such that the observation times are

$$t_j = t_1 + k_j h, j = 1\dots m. \quad (23)$$

In practice, the observation times are rounded to fit on this grid. Note that with this notation, $t_j = s_{k_j}$. We now proceed through a series of transformations to rewrite this constrained optimization problem into an unconstrained one.

First, we replace the constraints on x by a finite number of constraints as follows:

$$\begin{cases} f \in H, \text{ the RKHS with kernel } K, \\ x(s_{l+1}) = x(s_l) + \int_{s_l}^{s_{l+1}} f(x(s))ds \\ l = 0 \dots k-1. \end{cases} \tag{24}$$

Second, we discretize the constraints using the Euler method of integration:

$$\begin{cases} f \in H, \text{ the RKHS with kernel } K, \\ x(s_{l+1}) = x(s_l) + hf(x(s_l)) \\ \text{for } l = 0 \dots k-1. \end{cases} \tag{25}$$

Third, we replace the constrained optimization problem by an unconstrained one using a single Lagrange constant $\gamma > 0$. Notate $z_l = x(s_l)$, $l = 0 \dots k$,

$$\min_{z \in \mathbb{R}^{d(k+1)}, f \in H} J(z, f, \gamma), \tag{26}$$

with

$$J(z, f, \gamma) = \frac{1}{m} \sum_{j=1}^{m} |y_j - z_{k_j}|^2 + \gamma \frac{1}{k} \sum_{l=0}^{k-1} \left| z_{l+1} - z_l - hf(z_l) \right|^2 + \lambda ||f - f_0||_H^2. \tag{27}$$

It is instructive to remark the similarities between the loss function in equation (27) and the loss proposed in Physics-informed Neural Networks [17], where the observations are generated from an unknown partial differential equation. Indeed, the total loss function in Physics-informed Neural Networks can be decomposed as a sum of two functions: One that measures the deviation of solution from the observations, and the second usually defined as the residual function term, measures the violation of the partial differential equation constraint that the solution must satisfy. In our context,

$$\frac{1}{m} \sum_{j=1}^{m} |y_j - z_{k_j}|^2$$

corresponds to first function, and

$$\frac{1}{k} \sum_{l=0}^{k-1} \left| z_{l+1} - z_l - hf(z_l) \right|^2$$

corresponds to the residual function term. However there are some notable differences. In physics informed neural networks, the form of the PDE is known up to finite-dimensional parameters. The loss is viewed as a function of the solution to the partial differential equation and these finite dimensional parameters. The solution itself is modeled by a neural network. In our case, the loss is viewed as a function of the vector field and the initial solution. The differential equation is therefore characterized by the RKHS, usually infinite-dimensional. Moreover, equation (27) contains a regularization term penalizing vector fields with large RKHS norm, which is typical of loss function parametrized by RKHS functions.

2.5. Penalty method

The penalty method is an iterative method that consists of enforcing the constraints by increasing a penalty parameter, in this case γ. The schematic of the method is presented in Algorithm 1. At each step, the functional $J(z, f, \gamma)$ in (27) is minimized with respect to (z, f), for a fixed value of γ. Then, γ is increased. The optimization for (z, f) is done asynchronously, first optimizing over z for a fixed f, then optimizing over f for the newly updated z.

Let us now describe these optimization steps in more detail. For a fixed γ and f, $J(z, f, \gamma)$ in (27) is non-convex in z due to the presence of $f(z_l)$. Therefore we replace f by its first-order Taylor expansion evaluated at the value $z_l^{(s)}$ obtained in the previous iteration s:

$$f(z_l) \approx f(z_l^{(s)}) + (z_l - z_l^{(s)})^T \nabla_{z_l} f(z_l^{(s)}) \tag{28}$$

Note that with this approximation, J is convex, quadratic, and sparse in z. This allows the use of an efficient linear solver for this minimization. The number of unknowns is $d(k+1)$.

Note on the non-autonomous case: When the vector field is time-dependent, the vector field is evaluated at points of the form $f(t_l, z_l)$. Notice that the t_l's are the time points of the grid, therefore fixed and known. Hence, the linearization in equation (28) is made only with respect to the space variable:

$$f(z_l, t_l) \approx f(z_l^{(s)}, t_l) + (z_l - z_l^{(s)})^T \nabla_{z_l} f(z_l^{(s)}, t_l) \tag{29}$$

Algorithm 1 Penalty method for ODE-RKHS.

Init: $h, \rho, \lambda, f^{(0)}, \gamma^{(0)}, s = 0$
while termination condition is not met **do**
  $z^{(s+1)} \leftarrow \arg\min_{z \in \mathbb{R}^{d(k+1)}} J(z, f^{(s)}, \gamma^{(s)})$
  $f^{(s+1)} \leftarrow \arg\min_{f \in H} J(z^{(s+1)}, f, \gamma^{(s)})$
  $\gamma^{(s+1)} \leftarrow \gamma^{(s)}(1+\rho)$
  $s = s + 1$
  Check termination condition
end while

For a fixed γ and z, minimizing J in f is equivalent to a multivariate kernel ridge regression problem. After the change of variable, $g = f - f_0$, and setting

$$u_l = (z_{l+1} - z_l)/h - f_0(z_l), l = 0 \dots k-1, \tag{30}$$

we use the representer theorem to show that the minimizer in $f \in H$ of J is of the form

$$f(z) = f_0(z) + \sum_{l=0}^{k} K(z, z_l) w_l, \tag{31}$$

where $w_l \in \mathbb{R}^d$. Let $W = (w_1^T, \dots, w_{k+1}^T)$, be of dimension $(d(k+1), 1)$ and similarly let $U = (u_1^T, \dots, u_{k+1}^T)$ and K be the matrix with (d, d) block element $K_{kl} = K(x_k, x_l)$. We find that W is a minimizer of the convex quadratic function

$$\frac{\gamma h^2}{k} |U - KW|^2 + \lambda W^T K W \tag{32}$$

and thus W is the solution to the linear system:

$$\left(K + \frac{\lambda k}{\gamma h^2} I \right) W = U \tag{33}$$

The schematic algorithm is provided in Algorithm 1.

2.6. Initial condition and termination criteria

Since the algorithm will converge to a local minimum of the cost function, the choice of the initial condition is important. We use a gradient matching method.

1. Approximate the time derivatives of x at the observed times $\dot{x}(t_j)$, denoted $\hat{\dot{x}}(t_j)$
2. Estimate $f_0 \in H$ using ridge regression, i.e. minimize over H

$$G(f_0) = \frac{1}{m} \sum_{j=1}^{m} |\hat{\dot{x}}(t_j) - f_0(y_j)|^2 + \lambda ||f_0||_H^2 \tag{34}$$

There are several possibilities for the approximation in the first step depending on the sparsity of the data and the amount of noise. In the experiments below, we use central differences.

The termination condition of Algorithm 1 includes a fixed number of iterations S and a threshold on the quantity $||f^{(s+1)} - f^{(s)}||/||f^{(s)}||$ which allows for early stopping.

2.7. Multiple trajectories

We present here the extension of the method to multiple trajectories, say $n > 1$ subjects. We assume the same number of observations for each subject and regular sampling to simplify the presentation.

First, we replace (27) and (24) with

$$\min_{x,f} \frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} |y_{ij} - x_i(t_{ij})|^2 + \lambda ||f - f_0||_H^2, \tag{35}$$

under the constraints

$$\begin{cases} f \in H, \text{ the RKHS with matrix-valued kernel K,} \\ x_i(t) = x_i(t_1) + \int_{t_1}^{t} f(x_i(s)) ds, \\ \text{for } t_1 \le t \le t_m, i = 1 \dots n \end{cases} \tag{36}$$

We then proceed along the same steps as for the single trajectory case, leading to the unconstrained optimization problem, generalizing (26) and (27).

Algorithm 2 Multi Trajectories Penalty method for ODE-RKHS.

1: Init: $h, \rho, \lambda, f^{(0)}, \gamma^{(0)}, s = 0$
2: **while** termination condition is not met **do**
3: **for** $i = 1 \dots n$ **do**
4: $z_i^{(s+1)} \leftarrow \arg\min_{z_i \in \mathbb{R}^{d(k+1)}} J_{\text{multi}}(z, f^{(s)}, \gamma^{(s)})$
5: **end for**
6: $f^{(s+1)} \leftarrow \arg\min_{f \in H} J_{\text{multi}}(z^{(s+1)}, f, \gamma^{(s)})$
7: $\gamma^{(s+1)} \leftarrow \gamma^{(s)}(1+\rho)$
8: $s = s + 1$
9: Check termination condition
10: **end while**

Notate $z_{il} = x_i(s_l)$, $l = 0 \dots k$, $i = 1 \dots n$, and $z = (z_1, \dots, z_n)$

$$\min_{z \in \mathbb{R}^{nd(k+1)}, f \in H} J_{\text{multi}}(z, f, \gamma), \tag{37}$$

with

$$J_{\text{multi}}(z, f, \gamma) = \frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} |y_{ij} - z_{ik_j}|^2 + \gamma \frac{1}{nk} \sum_{i=1}^{n} \sum_{l=0}^{k-1} \left| z_{i,l+1} - z_{il} - hf(z_{il}) \right|^2 + \lambda ||f - f_0||_H^2. \tag{38}$$

The key point is that J_{multi} decouples the trajectories such that the optimization over z can be carried out separately for each trajectory. However, all the observations contribute to the estimation of f. The algorithm is presented in Algorithm 2. In Line 6: we use the no-trick formulation using Gaussian quadrature Fourier features as described in [18].

2.8. Computational complexity

We analyze the complexity of the Algorithm 2. The key parameters are:

1. d: the dimension of the observed vectors;
2. n: the number of observed trajectories;
3. k: the number of samples in the discretization of the time interval;
4. S: the number of steps in Algorithm 2;
5. n_F: the number of Fourier features.

We use $O(p^3)$ for the time complexity of solving a (dense) linear system with p variables and $O(w^2 p)$ in the case of a band matrix of width w, see [19]. Algorithm 2, line 4 consists in solving a linear system of size dk with a band matrix of bandwidth $w = 3d$, thus $O(kd^3)$ computations. Line 6 consists in solving d full linear systems of dimension n_F, thus $O(dn_F^3)$ computations. In total, we find $O(Snkd^3 + Sdn_F^3)$. Note that k is typically chosen proportional to the average number of data points per trajectory. Thus, overall, the algorithm is linear in the number of observations but cubic in the dimension of the observations.

2.9. Non autonomous systems, covariates, and irregular sampling

Non autonomous systems and covariates are handled by modifying the kernel. The issue of irregular sampling is addressed by replacing the first term of (27) by

$$\frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{m_i} (t_{i,j+1} - t_{ij}) |y_{ij} - z_{ik_j}|^2 \tag{39}$$

with $t_{i,m_i+1} = T$, $i = 1 \dots, n$

3. Consistency of the solution: a finite sample result

In this section, we assume that the algorithm solves the following optimization problem (where $t_{m+1} = T$ by definition):

$$\min_{\mathbb{R}^{d(k+1)}, f \in H} \sum_{j=1}^{m} (t_{j+1} - t_j) |y_j - z_{k_j}|^2, \tag{40}$$

Under the constraints:

1. $||f - f_0||_H \le R$, $|z_0| \le r$
2. $z_{l+1} = z_l + hf(z_l)$, $0 \le l \le k$

Notice that constraint 2 corresponds to the Euler method for the ODE: $\dot{x} = f(x)$. Therefore, by linearly interpolating between the times of subdivision s_l, $0 \leq l \leq k$, we can generate a solution $\hat{x}(\cdot)$ defined on $[0,T]$. We denote by $x^*(\cdot)$ the true trajectory generating the noisy observations y_j at each time t_j. The purpose of this section is to present a result controlling (in probability) the L^2 norm squared of $\hat{x} - x^*$:

$$||\hat{x} - x^*||_{L^2}^2 := \int_0^T |(\hat{x}(t) - x^*(t))|^2 dt \tag{41}$$

Let us make the following assumptions:

- $\mathbf{A_1}$: There exists an $f^* \in H, ||f^* - f_0||_H \leq R$ and $|x_0^*| \leq r$ such that $x^*(0) = x_0^*$ and $\dot{x}^*(t) = f^*(x^*(t))$ for every $0 \leq t \leq T$.
- $\mathbf{A_2}$: The noise variables ϵ_{ij} are independent and bounded in absolute value by a constant M_ϵ. (We can assume that the variables are subgaussian instead of bounded if we want to generalize this result)
- $\mathbf{A_3}$: The kernel K is $C^2(\mathbb{R}^d)$ in its first argument (this implies that it is also $C^2(\mathbb{R}^d)$ in its second argument).
- $\mathbf{A_4}$: The kernel K satisfies (13).

We refer to section 2.3 for examples of kernels satisfying $\mathbf{A_3}$ and $\mathbf{A_4}$.
These assumptions are sufficient for obtaining the main theorem of this section, controlling $||\hat{x} - x^*||_{L^2}^2$ with high probability.

Theorem 1. *Assuming $\mathbf{A_1}, \mathbf{A_2}$, $\mathbf{A_3}$ and $\mathbf{A_4}$, there exist positive constants*
K_1, K_2, K_3 and K_4, depending only on R, r, T, M_ϵ, N_K and the kernel K such that for every $\epsilon > 0$, with probability less than $\exp\left(\frac{-K_2\epsilon^2}{d\sum_{j=1}^m (t_{j+1}-t_j)^2}\right)$:

$$||\hat{x} - x^*||_{L_2}^2 \geq K_1 d \sqrt{\sum_{j=1}^m (t_{j+1} - t_j)^2} + h^2 K_3 d + K_4 d \sum_{j=1}^m (t_{j+1} - t_j)^2 + \epsilon. \tag{42}$$

For a better understanding of Theorem 1, assume a regular sampling of the interval $[0,T]$ with m points, so that for every j, $t_{j+1} - t_j = \frac{1}{m}$. In that case, under the same hypothesis, for any $\epsilon > 0$, with probability less than $\exp\left(\frac{-K_2 m\epsilon^2}{d}\right)$:

$$||\hat{x} - x^*||_{L_2}^2 \geq \frac{K_1 d}{\sqrt{m}} + \frac{K_4 d}{m} + h^2 K_3 d + \epsilon. \tag{43}$$

A proof of Theorem 1 is provided in the appendix. We provide here a description of the main ideas. The third term in the right hand side of inequality (42) corresponds to the global truncation error between the numerical solution of the ODE and the true solution. The second term corresponds to the error between $||\hat{x} - x^*||_{L^2}^2$ and $\frac{1}{m}\sum_{j=1}^m |x^*(t_j) - \hat{x}(t_j)|^2$. The first term is the leading term, assuming that h is always less than $\frac{1}{m}$. Assume that $\hat{x}$ solves the continuous-constraints optimization problem (without an Euler approximation), i.e.:

$$\min_{x,f} \frac{1}{m} \sum_{j=1}^m |y_j - x(t_j)|^2, \tag{44}$$

Under the constraints: $||f - f_0||_H \leq R$, $|x_0| \leq r$ and $x(t) = x_0 + \int_0^t f(x(u))du$, $\forall 0 \leq t \leq T$, we can then consider the "generalization" error:

$$\frac{1}{m} \sum_{j=1}^m |x^*(t_j) - \hat{x}(t_j)|^2. \tag{45}$$

An upper bound of this error is given by the first term. The main tool used to obtain the upper bound is Dudley's chaining inequality, see [20]. We notice that for every $i = 1, \ldots, d$, the set of coordinate functions x_i, where x and f satisfy the constraints of the continuous problem, is included in a set of functions that are uniformly Lipschitz continuous and bounded (the Lipschitz constant and bound do not depend on x_0 and f). Upper bounds of covering numbers of such functions are well-known, see [20], hence the use of Dudley's inequality.

One can easily transform the inequality on the probability of Theorem 1 to an inequality on $\mathbb{E}\left(||\hat{x} - x^*||_{L_2}^2\right)$. Indeed, let us assume for simplicity a regular sampling of m points the interval $[0,T]$. We denote by:

$$\hat{E}_{L_2} := ||\hat{x} - x^*||_{L_2}^2 - \frac{K_1 d}{\sqrt{m}} - \frac{K_4 d}{m} - h^2 K_3 d. \tag{46}$$

Using Theorem 1, we have the following inequality:

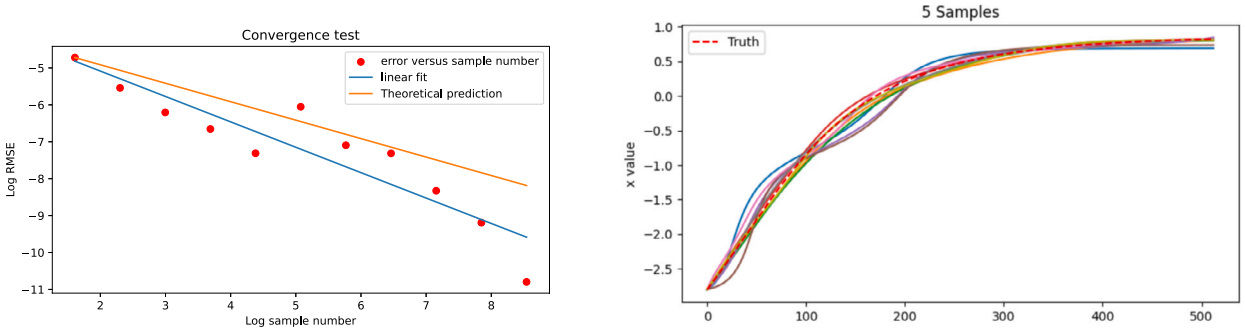


Fig. 2. On the left we plot the log of the average L_2 squared error between the true trajectory and the estimated one as a function of the log of the number of samples. A linear regression yields a slope of $-.8$ indicating convergence at a rate between $\frac{1}{\sqrt{m}}$ and $\frac{1}{m}$. On the right we plot the predicted trajectories when we use have 5 observations, together with the true trajectory (in the dotted line).

$$\mathbb{E}\left(|\hat{E}_{L_2}|\right) = \int_0^\infty \mathbb{P}\left(|\hat{E}_{L_2}| \geq \epsilon\right) \tag{47}$$

$$\leq \int_0^\infty \exp\left(\frac{-K_2 m\epsilon^2}{d}\right) \tag{48}$$

$$= \sqrt{\frac{\pi}{4K_2}}\sqrt{\frac{d}{m}} \tag{49}$$

This implies the following result.

Corollary 1. *Assume we have a regular sampling of m points on the interval $[0, T]$. Then:*

$$\mathbb{E}\left(||\hat{x} - x^*||_{L_2}^2\right) \leq \sqrt{\frac{\pi}{4K_2}}\sqrt{\frac{d}{m}} + \frac{K_1 d}{\sqrt{m}} + \frac{K_4 d}{m} + h^2 K_3 d. \tag{50}$$

To illustrate the inequality in (50), we conducted a simple toy experiment where the conditions of the theorem are satisfied, and evaluated the convergence rate. In this experiment we considered a one-dimensional autonomous system. We randomly initialized the weights of a function determined by 200 Fourier random features, recorded the norm of the function, and generated a trajectory of 5120 samples using this function. Then we took ten independent and identically distributed random samples of noise with a standard deviation of .05. This provided us with 10 noisy trajectories of 5120 samples (of the same trajectory but different samples of noise). Finally, we sub-sampled each of these ten noisy trajectories to get 2560 samples, 1280 samples,... all the way down to 5 samples. This gave us 10 training sets, each with 5, 10, 20, 40,..., 5120 samples. We trained the algorithm on each of these datasets and reported the average L_2 (squared) error between the estimated trajectory and the true one over the ten trajectories at each level of sparsity. In Fig. 2, we provide a plot of the log of the average L_2 (squared) errors as a function of the log of the number of samples used during training. Equation (50) predicts a slope at least $-.5$. We fit the data to a line of slope $-.8$, consistent with (50). We provide a plot with a line of slope $-.5$ for comparison.

4. Experiments

We report experiments for simulated data as well as for real data. In each case, we compare the performances of the proposed algorithm, generically named ODE-RKHS, with six other algorithms. This section is organized as follows: In subsection one, we present the various benchmark methods used for comparison. In subsection two, we present the tuning of the hyperparameters for the ODE-RKHS method. In subsection three, we describe fifteen simulated datasets and an example medical dataset. Finally, in subsection four, we report and comment on the performance of the ODE-RKHS method compared with the benchmark methods on all the datasets.

4.1. Benchmark methods

These algorithms constitute, up to our knowledge, the current state of the art for learning nonparametric ODEs from noisy data. We briefly review these algorithms and provide references below.

1. *Nonparametric Ordinary Differential Equations:* Nonparametric Ordinary Differential Equations (npODE) is presented in [12]. The authors use a Bayesian model with Gaussian processes (GP). It is the Bayesian counterpart of the frequentist model presented in

this paper. Unlike GP regression where the optimization can be computed in closed form, an approximate optimization method is required. The authors use inducing points, see [21] and sensitivity equations, see [22]. The npODE code was downloaded from http://www.github.com/cagatayyildiz/npode in February 2021. Given the normalized trajectory sets, we ran the algorithm with a scale factor of 1 and an ℓ_0 of 1. For the 2D systems, we used a width of the inducing point grid $W = 6$, matching the demonstration examples. For the 6D Lorenz96, we encountered out-of-memory errors for $W > 2$, possibly indicating an empirical scaling issue with the method. We thus used $W = 2$ for this system.

2. *Sparse Identification of Nonlinear Dynamics (Fourier and Polynomial Candidate Functions):* Sparse Identification of Nonlinear Dynamics (SINDy) is a highly cited technique for identifying nonlinear dynamics from data, see [4]. SINDy predicts governing dynamics equations using gradient matching via sparse regression. In the experiments shown, we test SINDy with two different libraries of possible functions: polynomials up to order three and Fourier features. We choose the SR3 sparsity regularization for its superior performance, detailed in [23], which has a threshold value as a hyperparameter. Other hyperparameters in our tests include the polynomial library's degree and the size and lengthscale of the Fourier features library. A grid search tuner was employed to determine the best hyperparameter values, with the same holdout and evaluation sets as in the competing algorithms. pySINDy v1.6.3 was used for the implementation [24]. We use the AutoKoopman library to tune the hyperparameters, described in [25].
3. *Extended Dynamic Mode Decomposition:* The Koopman operator is an infinite dimensional linear operator that captures the dynamics of a non-linear dynamical system. Dynamic Mode Decomposition (DMD), described in [10], can approximate the Koopman operator's eigenvalues and eigenvectors based on observations of the system state. Extended DMD (EDMD) generalizes to nonlinear systems learning by approximating the Koopman operator in a high-dimensional space of observables, see [26]. These observables must be selected before using EDMD, and can be chosen ad-hoc or by using library learning methods [27]. We use random Fourier features as the observable functions for these experiments, specified in [28]. We use the AutoKoopman library to tune the hyperparameters via Bayesian optimization, available at https://github.com/EthanJamesLew/AutoKoopman.
4. *Kernel Analog Forecasting:* Analog forecasting is a time series prediction method that utilizes the idea of analog forecasting that follows the evolution of a historical time series that most closely matches the current state. Kernel analog forecasting (KAF) replaces single-analog forecasting with weighted ensembles of analogs constructed using local similarity kernels that employ several dynamics-dependent features designed to improve forecast skill [29] [30]. Our KAF implementation is based on https://github.com/rward314/StreamingKAF. Hyperparameters are the kernel function and rank used for the number of eigenvalues found from the data-defined kernel matrix. We selected a Gaussian kernel and grid tuned for rank and kernel lengthscale. We use the same eigenvalue multiplier of 10^{-4} as the referenced code.
5. *Sparse Cyclic Recovery:* We implement the method formulated in [31] which is well-suited for the experiments as it is designed for learning structured dynamical systems from under-sampled and possibly noisy state-space measurements. For index invariant systems, the method generates cyclic permutations to augment the training data. Then, it builds a library of Legendre polynomials of candidate functions and does basis pursuit with thresholding to recover the dynamics. The hyper-parameters involved are the parameters for the Douglas-Rachford algorithm used to solve the Legendre basis pursuit (L-BP) problem and the Legendre polynomial degree; we tune these parameters via grid search. We referenced the parameters used in their GitHub project https://github.com/linanzhang/SparseCyclicRecovery. We utilize the same candidate functions as the paper, but tune the noise threshold σ and the μ, τ parameters of the optimizer. Because of compute effort limitations, we set the maximum number of optimization iterations to 10^4.

4.2. Validation, initialization, and selection of hyper-parameters in the ODE-RKHS algorithm

We use the Multi Trajectories Penalty method for ODE-RKHS described in Algorithm 2, and a random Fourier features kernel. For each coordinate, we chose a bandwidth equal to 20% of the range of the data. We set $\gamma = 1$ and fit λ, ρ using a validation set consisting of 20 percent of the training data. We set a maximum of $S = 500$ iterations and used the early stopping criterion of stopping when the ratio $||f^{(s+1)} - f^{(s)}||/||f^{(s)}||$ was less than 10^{-3}. Initialization of f_0 was done via gradient matching, see section 2.6.

4.3. Datasets

We ran experiments with the same training, validation and test sets for all the algorithms. Testing consisted of computing predicted trajectories starting at the initial condition of each test trajectory.

4.3.1. Oscillator data

The FitzHugh-Nagumo (FHN) oscillator data is a controlled experiment with known and easy-to-visualize 2D trajectories. It has helped calibrate the algorithm described in this paper. It was also demonstrated in [12] for the npODE algorithm. We ran experiments using a simulated dataset generated as follows:

$$\begin{aligned} \dot{v} &= v - v^3/3 - w + 1 \\ \dot{w} &= 0.08(v + 0.7 - 0.8w) \end{aligned} \tag{51}$$

Intermediate and final results of the ODE-RKHS algorithm are presented in Fig. 3 for the FHN data. Notice that during the first steps, shown on the top line, the estimated trajectories with solid color lines are rough but fit the data closely. During the later steps, shown on the bottom line, the trajectories are smoother but still fit the data.

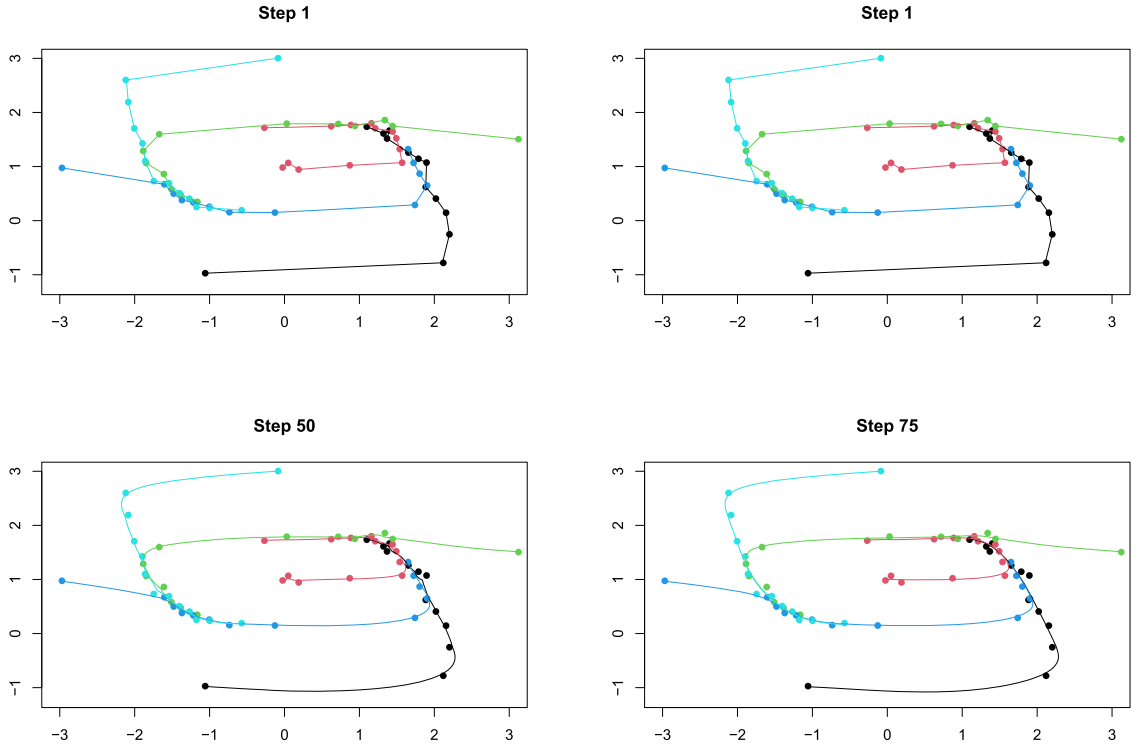


Fig. 3. Illustration of the ODE-RKHS Algorithm: The dots show the observations. The estimated trajectories are shown with lines and curves with corresponding colors. Steps $i = 1,25,50$, and 75 are shown from left to right and from top to bottom.

We generated a set of 50 noiseless trajectories. There were 201 observations per trajectory, one for each .1 increment in time. To generate the training sets, we added samples of Gaussian noise to these fifty trajectories. There were five levels of noise, with respective standard deviations $\sigma \in \{0.120, 0.365, 0.610, 0.855, 1.100\}$. We generated a single test set of 100 trajectories without noise, again with 201 observations per trajectory separated by .1 time increments.

4.3.2. *Lorenz63 data*

Our next experiment was on the Lorenz system defined by the equations

$$\begin{aligned} \dot{x} &= 10(y - x) \\ \dot{y} &= x(28 - z) - y \\ \dot{z} &= xy - \frac{8}{3}z \end{aligned} \tag{52}$$

We generated 50 noiseless trajectories with 201 observations per trajectory, each separated by a 0.01 increment in time. Next, we generated samples of Gaussian noise with levels $\sigma \in \{0.5, 1.2, 1.9, 2.6, 3.3\}$. We added the noise samples to the noiseless trajectories to generate five training sets. Then we generated a single test set consisting of 100 trajectories, each with 201 observations at 0.01 time increments.

4.3.3. *Lorenz96*

The Lorenz96 data arises from [32]. The chaotic system is defined for $n = 6$ dimension by:

$$\dot{x}_k = -x_{k-1}x_{k-1} + x_{k+1}x_{k-1} - x_k + F, k = 1 \dots 6 \tag{53}$$

We have selected $F = 8$. Indices wrap-around so that $x_{-1} = x_6$ and $x_7 = x_1$. To construct the training set, we generated a set of 100 noiseless trajectories, each with 100 observations. The observations were separated by a time increment of 0.01. We added the five levels of Gaussian noise to the noiseless data to generate five different training sets. The standard deviations of the noise generated here are $\sigma \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$. Our test set consisted of 150 noiseless trajectories, each with 100 points on them. The time increment between observations was the same as the training set.

4.3.4. *The accumulation of Amyloid in the cortex of aging subjects*

The accumulation of Amyloid in the brain is believed to be one of the earliest pathological mechanisms of Alzheimer's disease, beginning more than a decade before the onset of clinical symptoms, see [33].

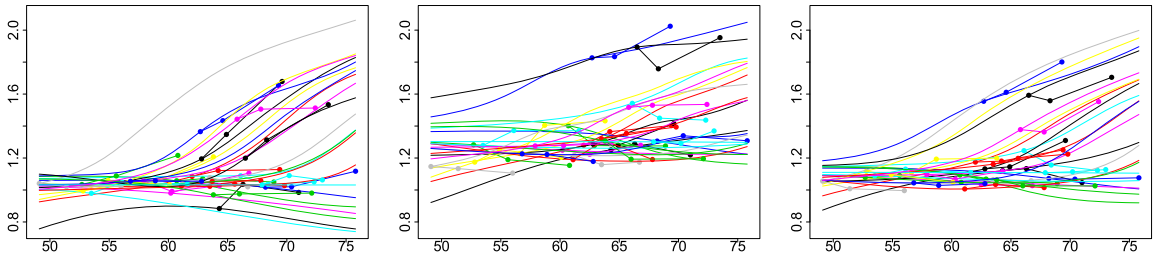


Fig. 4. Amyloid prediction experiment. Horizontal axis is in years. Vertical axis corresponds to DVR. The left-most image corresponds to the gyrus rectus, the middle to the cingulum and the right to the precuneus.

Table 1
Results for Amyloid data. Minimum errors are in bold.

	Err
npODE	.59
KAF	.84
Koopman	.52
L-BP	.40
ODE-RKHS	**.36**
SINDy Fourier	.42
SINDy Polynomial	.39

Based on observations from several longitudinal Amyloid positron emission tomography (PET) studies, it is believed that the rate of Amyloid accumulation is closely associated with the level of Amyloid at the same age, see [34]. We develop a principled mathematical model capturing this phenomenon and use it to predict the accumulation of Amyloid across individuals longitudinally.

We used (PiB) PET scans from the Wisconsin Registry for Alzheimer's Prevention (WRAP) to assess global Amyloid burden, measured by the Distribution Volume Ratio (DVR).[1] The number of subjects in this study is $n = 179$, with 3.06 visits on average, over an average span of 6.84 years. We fit the model in (11) to the posterior cingulum, precuneus and gyrus rectus DVRs, averaging the left and right DVR in each case. These regions are known to show Amyloid accumulation early in the disease process. Fig. 4 provides a visualization of the trajectories estimated using RKHS-ODE super-imposed (same color) with the data. This shows that the estimated trajectories are qualitatively accurate.

4.4. Evaluation

Testing consisted of computing predicted trajectories starting at the initial condition of the test trajectories and computing the following error measurement for each predicted trajectory:

$$\text{Err} := \sqrt{\sum_{i=2}^{n} (t_i - t_{i-1}) \|y_i - \hat{y}_i\|^2} \tag{54}$$

where t_i is the i^{th} observation time, y_i is the i^{th} observation of the test trajectory, $\hat{y}_i$ is the i^{th} point of the predicted trajectory and n is the number of observations in the trajectory. For each dataset, we report the average error measurement over the test set trajectories.

In Table 1, we report the performance of the ODE-RKHS method and other benchmark methods on the Amyloid dataset. The average L_2 norm errors (Err) between predicted Amyloid level trajectories and true level trajectories are reported. The ODE-RKHS algorithm yields the lowest average L_2 error among the seven compared methods.

In Table 2, we report the performance of ODE-RKHS and the benchmark methods on the 3 simulated datasets (FHN, Lorenz63, and Lorenz96) with the 5 simulated levels of noise, level 1 corresponding to the noise with the smallest standard deviation. **The ODE-RKHS algorithm performed best in 10 out of the 15 simulated test sets.** The second best performing method was SINDy Polynomial with the lowest error in just 2 out of the 15 simulated datasets. Moreover, the 2 cases where SINDy polynomial performed best correspond to the lowest noise levels of the Lorenz63 dataset, indicating that our method is more robust to higher noise levels.

5. Discussion

We proposed an algorithm for learning non-parametric ODEs assuming that the function f generating the vector field in $\mathbb{R}^d$ belongs to a vector-valued RKHS with a kernel satisfying certain regularity conditions. The data input of the algorithm consists of

[1] The data used for this experiment has been obtained from the Wisconsin Registry for Alzheimer's Prevention. See https://wrap.wisc.edu/. A request for accessing this data can be initiated from this website.

Table 2
Performance Table for the 3 Simulated Datasets.

FHN Noise Level 1		Lorenz63 Noise Level 1		Lorenz96 Noise Level 1	
	Err		Err		Err
npODE	1.53	npODE	17.49	npODE	1.61
KAF	6.019	KAF	22.75	KAF	2.18
Koopman	2.27	Koopman	5.96	Koopman	**.25**
L-BP	5.55	L-BP	16.35	L-BP	1.02
ODE-RKHS	**.53**	ODE-RKHS	9.06	ODE-RKHS	.30
SINDy Fourier	5.59	SINDy Fourier	23.37	SINDy Fourier	.52
SINDy Polynomial	1.28	SINDy Polynomial	**2.18**	SINDy Polynomial	1.13
FHN Noise Level 2		Lorenz63 Noise Level 2		Lorenz96 Noise Level 2	
	Err		Err		Err
npODE	1.57	npODE	18.75	npODE	1.29
KAF	8.35	KAF	22.34	KAF	2.17
Koopman	3.15	Koopman	13.67	Koopman	1.09
L-BP	6.87	L-BP	18.33	L-BP	1.10
ODE-RKHS	**1.16**	ODE-RKHS	11.24	ODE-RKHS	**.42**
SINDy Fourier	5.54	SINDy Fourier	21.63	SINDy Fourier	.84
SINDy Polynomial	2.60	SINDy Polynomial	**10.73**	SINDy Polynomial	1.76
FHN Noise Level 3		Lorenz63 Noise Level 3		Lorenz96 Noise Level 3	
	Err		Err		Err
npODE	3.07	npODE	20.06	npODE	1.31
KAF	8.25	KAF	21.96	KAF	2.16
Koopman	3.57	Koopman	16.12	Koopman	1.11
L-BP	5.48	L-BP	19.63	L-BP	1.17
ODE-RKHS	**1.83**	ODE-RKHS	**13.38**	ODE-RKHS	**.52**
SINDy Fourier	6.50	SINDy Fourier	22.88	SINDy Fourier	1.00
SINDy Polynomial	2.84	SINDy Polynomial	15.88	SINDy Polynomial	1.23
FHN Noise Level 4		Lorenz63 Noise Level 4		Lorenz96 Noise Level 4	
	Err		Err		Err
npODE	4.33	npODE	19.61	npODE	1.95
KAF	8.53	KAF	21.82	KAF	2.16
Koopman	7.18	Koopman	17.92	Koopman	1.02
L-BP	6.58	L-BP	21.03	L-BP	1.09
ODE-RKHS	**2.20**	ODE-RKHS	**14.38**	ODE-RKHS	**.83**
SINDy Fourier	9.47	SINDy Fourier	22.07	SINDy Fourier	1.24
SINDy Polynomial	5.57	SINDy Polynomial	20.67	SINDy Polynomial	3.03
FHN Noise Level 5		Lorenz63 Noise Level 5		Lorenz96 Noise Level 5	
	Err		Err		Err
npODE	4.37	npODE	**19.45**	npODE	2.10
KAF	7.62	KAF	21.49	KAF	2.15
Koopman	7.12	Koopman	18.97	Koopman	1.23
L-BP	7.51	L-BP	21.68	L-BP	1.34
ODE-RKHS	**1.97**	ODE-RKHS	21.20	ODE-RKHS	1.23
SINDy Fourier	12.29	SINDy Fourier	22.19	SINDy Fourier	**1.18**
SINDy Polynomial	9.00	SINDy Polynomial	23.23	SINDy Polynomial	1.67

Minimum values are in bold. ODE-RKHS performs best in 10 out the 15 datasets.

noisy observations at different times of multiple trajectories. The algorithm is linear in the number of observations but cubic in their dimension. We proved the consistency of the estimated trajectory, showing that the L^2 squared distance between the estimated trajectory and the true one vanishes as more observations are collected. We assessed the algorithm with simulated and real data and obtained results that consistently compare favorably with the state of the art on a wide range of noise levels.

CRediT authorship contribution statement

Kamel Lahouel: Conceptualization, Formal analysis, Investigation, Methodology, Supervision, Writing – original draft, Writing – review & editing. **Michael Wells:** Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Victor Rielly:** Conceptualization, Data curation, Formal analysis, Methodology, Software, Validation, Writing – original draft, Writing – review & editing. **Ethan Lew:** Data curation, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **David Lovitz:** Investigation, Methodol-

ogy, Software, Visualization, Writing – original draft. **Bruno M. Jedynak:** Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Supervision, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The code and the simulated data will be made available at the time of publication.

Acknowledgements

The work at Portland State University was partly funded using the National Institute of Health RO1AG021155, R01EY032284, and R01AG027161, National Science Foundation 2136228 and 2112085, and the Google Research Award "Kernel PDE" 7026026. The funding sources had no involvement in the study design; in the collection, analysis, and interpretation of data; in the report's writing; and in the decision to submit the article for publication. The material of Galois, Inc. is based upon work supported by the Air Force Research Laboratory (AFRL) and DARPA under Contract No. FA8750-20-C-0534. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s). They do not necessarily reflect the views of the Air Force Research Laboratory (AFRL) and DARPA.

Appendix A. Consistency of the estimator of the trajectory

A.1. Assuming we solve the problem without Euler approximation

This section gives the proof of the theorem presented in section 3 of the main text. We present the proof for $d = 1$ since the generalization to multiple dimensions is straightforward. We also present the proof for the case of autonomous systems. Keeping the notations of the main text, we make the following assumptions:

- $\mathbf{A_1}$: There exist an $f^* \in H, ||f^* - f_0||_H \leq R$ and $|x_0^*| \leq r$ such that $x^*(0) = x_0^*$ and $\dot{x}^*(t) = f^*(x^*(t))$ for every $0 \leq t \leq T$.
- $\mathbf{A_2}$: The noise variables ϵ_j are independent and bounded by a constant M_ϵ, with a variance denoted by σ^2. (We can assume that the variables are subgaussian instead of bounded if we want to generalize this result)
- $\mathbf{A_3}$: The kernel K is $C^2(\mathbb{R})$ in its first argument (this implies that it is also $C^2(\mathbb{R})$ in its second argument).
- $\mathbf{A_4}$: The kernel K satisfies the hypothesis of Lemma 1.

Without loss of generality, we will assume that $f_0 = 0$ in our proof.

Let H be the RKHS with reproducing kernel K. Let $f \in H$ such that $||f||_H \leq R$. We know using assumption $\mathbf{A_4}$ and Lemma 1 that f is uniformly Lipschitz, with a Lipschitz constant that does not depend on f that we denote by L_1. Specifically,

$$|f(x) - f(y)| \leq L_1 |x - y| \tag{A.1}$$

with $L_1 = N_K R$ Using (A.1), we will prove the following lemma:

Lemma 2. *Assuming* $\mathbf{A_4}$*, consider the set of solutions to the problem*

$$\frac{\partial x}{\partial t} = \dot{x} = f(x), x(t_0) = x_0 \tag{A.2}$$

where f belongs to the RKHS with kernel K, $|x_0| \leq r$ and $t \in [0, T]$. Then any solution x in this set of solutions is bounded by a uniform constant B_1 that only depends on T, R, L_1 and $L_3^2 := \sup_{||x|| < C} |K(x, x)|$.

Specifically,

$$|x(t) - x(t_0)| \leq B_1 = T L_3 R e^{L_1 T} \tag{A.3}$$

Proof. We start by taking f in our class of functions and x_0 such that $|x_0| \leq r$. We therefore can write:

$$|x(t) - x_0| = \left| \int_0^t (f(x(s)) - f(x_0)) ds + t f(x_0) \right| \tag{A.4}$$

$$\leq \int_0^t |f(x(s)) - f(x_0)| ds + t ||f||_H \sqrt{K(x_0, x_0)} \tag{A.5}$$

$$\leq L_1 \int_0^t |x(s) - x_0| ds + T L_3 R \tag{A.6}$$

Now denote by $G(t) := |x(t) - x_0|$. If we prove that $G(t)$ is bounded by a constant depending only on T, R, L_1 and L_3, we will be done. So far we have:

$$G(t) \leq L_1 \int_0^t G(s) ds + T L_3 R \tag{A.7}$$

Denote by $V(t) := \int_0^t G(s) ds$. We have that:

$$V'(t) \leq L_1 V(t) + T L_3 R \tag{A.8}$$

which implies:

$$e^{-L_1 t} V'(t) - L_1 e^{-L_1 t} V(t) \leq T L_3 R e^{-L_1 t} \tag{A.9}$$

Integrating the inequality between 0 and t using the fact that $V(0) = G(0) = 0$, we obtain:

$$\exp(-L_1 t) V(t) \leq \frac{T L_3 R}{L_1} (1 - e^{-L_1 t}) \tag{A.10}$$

or, equivalently,

$$V(t) \leq \frac{T L_3 R}{L_1} (e^{L_1 t} - 1) \tag{A.11}$$

Finally since $V'(t) = G(t) \leq L_1 V(t) + T L_3 R$, we have:

$$G(t) \leq T L_3 R e^{L_1 t} \leq T L_3 R e^{L_1 T} \quad \square \tag{A.12}$$

Let us now introduce the following notations:

- We denote by $x(x_0, f, t)$ the solution to the ODE with derivative f and initial condition x_0
- y_i is the observed noisy point from the trajectory at time t_i.
- $x^*(t)$ is the true trajectory evaluated at time t

We now proceed with the following reasoning. We assume that our trajectory minimizes

$$\hat{L}(f, x_0) := \sum_{i=1}^{m} (t_{i+1} - t_i) \left((x(x_0, f, t_i) - y_i)^2 - \sigma^2 \right) \tag{A.13}$$

over (f, x_0) such that $||f||_H \leq R$, and $|x_0| \leq r$. We denote the minimizer by $(\hat{f}, \hat{x}_0)$.

When x_0 and f are fixed and not data dependent (deterministic), the expected value of $\hat{L}(f, x_0)$ is:

$$L(f, x_0) := \sum_{i=1}^{m} (t_{i+1} - t_i)(x(x_0, f, t_i) - x^*(t_i))^2 \tag{A.14}$$

Notice that $\mathbf{A_1}$ implies:

$$min_{||f||_H \leq R, |x_0| \leq r} L(f, x_0) = L(f^*, x_0^*) = \sum_{i=1}^{m} (t_{i+1} - t_i)(x^*(t_i) - x^*(t_i))^2 = 0 \tag{A.15}$$

Our goal is to evaluate $L(\hat{f}, \hat{x}_0)$ and obtain a generalization bound. We have:

$$L(\hat{f}, \hat{x}_0) = L(\hat{f}, \hat{x}_0) - \hat{L}(\hat{f}, \hat{x}_0) + \hat{L}(\hat{f}, \hat{x}_0) - \hat{L}(f^*, x_0^*) + \hat{L}(f^*, x_0^*) - L(f^*, x_0^*) \tag{A.16}$$

And therefore, since the middle term in (A.16): $\hat{L}(\hat{f}, \hat{x}_0) - \hat{L}(f^*, x_0^*) < 0$,

$$L(\hat{f}, \hat{x}_0) \leq \sup_{||f||_H \leq R, |x_0| \leq r} 2 |L(f, x_0) - \hat{L}(f, x_0)| \tag{A.17}$$

We thus consider the following quantity:

$$\text{Err} := \sup_{||f||_H \leq R, |x_0| \leq r} |\hat{L}(f, x_0) - L(f, x_0)| \tag{A.18}$$

Expanding this quantity we get:

$$\sup_{||f||_H \leq R, |x_0| \leq r} \left| \sum_{i=1}^{m} (t_{i+1} - t_i)(y_i^2 - x^*(t_i)^2 - \sigma^2 - 2x(x_0, f, t_i)(y_i - x^*(t_i)) \right| \tag{A.19}$$

Notice that if we replace for a given single i, $y_i = x^*(t_i) + \epsilon_i$ by $\tilde{y_i} = x^*(t_i) + \tilde{\epsilon}_i$, the quantity of equation (A.19) will change by a quantity bounded by some constant $K_2(t_{i+1} - t_i)$, that we can bound by $4(B_1 + r + M_\epsilon)M_\epsilon + 4(B_1 + r)M_\epsilon$. Therefore, using McDiarmid inequality [35]:

$$\mathbb{P}\left(\text{Err} \geq \mathbb{E}(\text{Err}) + \epsilon\right) \leq \exp\left(\frac{-2\epsilon^2}{K_2^2 \sum_{i=1}^{m} (t_{i+1} - t_i)^2} \right) \tag{A.20}$$

We therefore need to provide an upper bound of $\mathbb{E}(\text{Err})$. For that, we are going to view:

$$|\hat{L}(f, x_0) - L(f, x_0)| = \left| \sum_{i=1}^{m} (t_{i+1} - t_i)(y_i^2 - x^*(t_i)^2 - \sigma^2 - 2x(x_0, f, t_i)(y_i - x^*(t_i))) \right| \tag{A.21}$$

as a stochastic process indexed by x, where $x \in \mathcal{X}$: Set of all solutions $x(f, x_0, .)$ for all $||f||_H \leq R$ and $|x_0| \leq r$. In other words, we view the process $|\hat{L}(f, x_0) - L(f, x_0)|$ indexed by f and x_0 as:

$$|\hat{L}(x) - L(x)| \tag{A.22}$$

where $x \in \mathcal{X}$ is some $x(f, x_0, .)$. Notice that Err is also:

$$\sup_{x \in \mathcal{X}} |\hat{L}(x) - L(x)| \tag{A.23}$$

Notice that x is a subset of continuous functions defined on $[0, T]$. Therefore we can equip $\mathcal{X}$ with the metric structure $(\mathcal{X}, ||.||_\infty)$. We will apply Dudley's inequality (see for e.g. [20], theorem 8.1.3) to bound:

$$\mathbb{E}(\text{Err}) = \mathbb{E}\left(\sup_{||f||_H \leq R, |x_0| \leq r} |\hat{L}(f, x_0) - L(f, x_0)| \right) \tag{A.24}$$

To apply Dudley's inequality, we are going to use the following lemma.

Lemma 3. *The solutions $x \in \mathcal{X}$ are Lipschitz with a Lipschitz constant that is uniform over $\mathcal{X}$, i.e., there exists a constant L_6 such that for every $x \in \mathcal{X}$, $t \in [0, T]$ and $s \in [0, T]$:*

$$|x(t) - x(s)| \leq L_6 |t - s| \tag{A.25}$$

K_6 depends on R, B_1, r and the kernel K.

Proof. Let x_0 such that $|x_0| \leq r$ and f such that $||f||_H \leq R$. We have:

$$|\dot{x}(x_0, f, t)| = |f(x(t))| \tag{A.26}$$

$$\leq R \sqrt{\sup_{|x| \leq B_1 + r} K(x, x)} \quad \square \tag{A.27}$$

As a consequence, if we denote by $\mathcal{N}(\mathcal{X}, \epsilon)$ the covering number of $\mathcal{X}$ with a radius ϵ we have the existence of a constant L_7 (L_7 only depends on B_1, r and L_6) such that:

$$\mathcal{N}(\mathcal{X}, \epsilon) \leq \exp\left(\frac{L_7}{\epsilon} \right), \tag{A.28}$$

where we used a known upper bound that can be found for example in [20] (exercise 8.2.7) on the covering number of uniformly bounded Lipschitz continuous functions defined on a finite interval.

Using this result combined with Dudley's inequality, we obtain the existence of a constant L_8 (depending only on L_7) such that:

Proposition 1.

$$\mathbb{E}(\text{Err}) \leq L_8 \sqrt{\sum_{i=1}^{m} (t_{i+1} - t_i)^2} \tag{A.29}$$

Proof. Apply Dudley's inequality to Err using inequality (A.28) and the fact that the diameter of $\mathcal{X}$ is finite bounded by $2(B_1 + r)$ and that for every $M < \infty$

$$\int_0^M \sqrt{\log\left(\mathcal{N}(\mathcal{X},\epsilon)\right)}d\epsilon \leq \int_0^M \sqrt{\log\left(\exp\left(\frac{K_7}{\epsilon}\right)\right)}d\epsilon < \infty \quad \square \tag{A.30}$$

As a consequence, using (A.20) and theorem (1), we obtain the following inequality:

$$\mathbb{P}\left(\text{Err} \geq L_8\sqrt{\sum_{i=1}^m (t_{i+1}-t_i)^2}+\epsilon\right) \leq \exp\left(\frac{-2\epsilon^2}{K_2^2\sum_{i=1}^m (t_{i+1}-t_i)^2}\right) \tag{A.31}$$

Using inequalities (A.17) and (A.31) we finally obtain the following theorem:

Theorem 2. *With assumptions* $\mathbf{A_1}, \mathbf{A_2}$, $\mathbf{A_3}$ *and* $\mathbf{A_4}$, *there exist constants* L_9 *and* K_2 *depending only on* R, r, T, M_ϵ *and the kernel* K *such that for every* ϵ:

$$\mathbb{P}\left(L(\hat{f},\hat{x}_0) \geq L_9\sqrt{\sum_{i=1}^m (t_{i+1}-t_i)^2}+\epsilon\right) \leq \exp\left(\frac{-2\epsilon^2}{K_2^2\sum_{i=1}^m (t_{i+1}-t_i)^2}\right) \tag{A.32}$$

A.2. Including the Euler approximation

In reality, the solution (trajectory) that we propose for every f and x_0 is not $x(x_0,f,.)$ the solution of the ODE but $\tilde{x}(x_0,f,h,.)$, the solution obtained with an Euler's method of time step h. The idea is to use the fact that under some sufficient conditions, we know how to bound the error between Euler's method and the true solution. For example, we know that if f is Lipschitz with a Lipschitz constant K_1 and the solution $x(x_0,f,.)$ is C^2 with a constant K_{11} such that:

$$x''(x_0,f,t) \leq L_{11}, \forall 0 \leq t \leq T \tag{A.33}$$

then we have the following global truncation error bound [36]:

$$\max_{1\leq i\leq n} |x(x_0,f,t_i)-\tilde{x}(x_0,f,h,t_i)| \leq \frac{hL_{11}}{2L_1}\left(\exp^{L_1 T}-1\right) \tag{A.34}$$

We already showed that f is Lipschitz with some constant L_1. To ensure the condition of inequality (A.33), notice that:

$$x''(x_0,f,t) = f(x(x_0,f,t))f'(x(x_0,f,t)) \tag{A.35}$$

Since we already showed that the solutions $x(x_0,f,.)$ are uniformly bounded by B_1+r, it is sufficient to ensure that f is C^1. This is true if we assume that our kernel K is C^2 and hence (A.34) will be insured.

Taking into account the Euler approximation and the error bound, the steps of the consistency proof are identical only with the following important difference in equation (A.15) from the previous section

$$\min_{||f||_H\leq R, |x_0|\leq r} L(f,x_0) \leq L(f^*,x_0^*) \tag{A.36}$$

with

$$L(f^*,x_0^*) = \sum_{i=1}^m (t_{i+1}-t_i)(\tilde{x}^*(t_i,h)-x^*(t_i))^2 \leq \frac{h^2 {L_{11}}^2 T}{4L_1^2}\left(\exp^{L_1 T}-1\right)^2 := L_{12} \tag{A.37}$$

With this modification, Theorem 2 becomes:

Theorem 3. *Assuming* $\mathbf{A_1}, \mathbf{A_2}$, $\mathbf{A_3}$ *and* $\mathbf{A_4}$, *there exist constants* K_2, L_{12} *and* L_{13} *depending only on* R, r, T, M_ϵ *and the kernel* K *such that for every* ϵ:

$$\mathbb{P}\left(L(\hat{f},\hat{x}_0) \geq L_{13}\sqrt{\sum_{i=1}^m (t_{i+1}-t_i)^2}+h^2L_{12}+\epsilon\right) \leq \exp\left(\frac{-2\epsilon^2}{K_2^2\sum_{i=1}^m (t_{i+1}-t_i)^2}\right) \tag{A.38}$$

A.3. L^2 squared distance between the true solution and the estimated trajectory

In reality $L(\hat{f},\hat{x}_0)$ is an approximation of the L^2 norm squared

$$||x(\hat{f},\hat{x}_0,\cdot)-x^*(\cdot)||_{L_2}^2 := \int_0^T \left(x(\hat{f},\hat{x}_0,t)-x^*(t)\right)^2 dt \tag{A.39}$$

Since we proved that the solutions are uniformly bounded by $(B_1 + r)$ and $\dot{x}$ is bounded by L_6, we have $t \to \left(x(\hat{f}, \hat{x}_0, t) - x^*(t)\right)^2$ is Lipschitz with Lipschitz constant $8(B_1 + r)L_6$ (we just bound the norm of the derivative). Therefore:

$$|||x(\hat{f}, \hat{x}_0, \cdot) - x^*(\cdot)||^2_{L_2} - L(\hat{f}, \hat{x}_0)| \leq 8(B_1 + r)L_6 \sum_{i=1}^{m} (t_{i+1} - t_i)^2 \tag{A.40}$$

Which proves Theorem 2 of the main text.

Appendix B. Kernels

We are interested in listing kernels that satisfy Lemma 1, and thus can be used to model ODEs admitting a single solution. There are cases when one can directly verify the hypothesis of Lemma 1. In the case of translation invariant kernels, one can use the Bochner theorem to provide a sufficient condition as explained in the next section.

B.1. Translation invariant kernels

We consider translation invariant scalar positive definite kernels over $\mathbb{R}^d$, that is kernels for which

$$k(u, v) = h(u - v), u, v \in \mathbb{R}^d \tag{B.1}$$

The Bochner theorem provides a characterization of translation invariant kernels. Specifically, there exists a probability density q with respect to the Lebesgues measure over $\mathbb{R}^d$ such that

$$h(x) = h(0) \int_{\mathbb{R}^d} e^{ix^T y} q(y) dy \tag{B.2}$$

Furthermore, since we restrict our attention to real-valued kernels,

$$h(x) = h(0) \int_{\mathbb{R}^d} cos(x^T y) q(y) dy \tag{B.3}$$

The gradient of h is then formally the vector of length d

$$\nabla h(x) = -h(0) \int_{\mathbb{R}^d} y sin(x^T y) q(y) dy \tag{B.4}$$

and the Hessian of h is formally the matrix

$$\nabla\nabla h(x) = -h(0) \int_{\mathbb{R}^d} (yy^T) cos(x^T y) q(y) dy \tag{B.5}$$

Translation invariant kernels that satisfy Lemma 1 are such that

$$Q(x) == c||x||^2 + 2(h(x) - h(0)) \geq 0 \tag{B.6}$$

for some constant $c > 0$ and for any $x, y \in \mathbb{R}^d$. Notice that $Q(0) = 0$. Next, since $\nabla h(0) = 0$, $\nabla Q(0) = 0$. Moreover,

$$\nabla\nabla Q(x) = 2cI + 2\nabla\nabla h(x) \tag{B.7}$$

where I is the identity matrix. Next, since $\nabla\nabla Q$ is a symmetric matrix, it has real eigenvalues. Suppose these eigenvalues are bounded uniformly from below. In that case, one can choose a constant c large enough such that $\nabla\nabla Q(x)$ is positive definite for each $x \in \mathbb{R}^d$ which implies that Q is convex and since $Q(0) = 0$ and $\nabla Q(0) = 0$, $Q(x) \geq 0$ for each $x \in \mathbb{R}^d$ and the conditions for Lemma 1 are satisfied. A sufficient condition for this to happen is that all the coordinates of $\nabla\nabla h$ are bounded, i.e., for each $i \in \{1, \ldots, d\}$, $E[Y_i^2] < \infty$, where Y_i is a random variable with density q_i, the i^{th} marginal of q.

B.2. Explicit kernels:

We begin by observing the condition

$$d^2_{K_{ii}}(u, v) \leq N_K^2 |u - v|^2, \forall u, v \in \mathbb{R}^d, i = 1, ..., d \tag{B.8}$$

is equivalent to the condition:

$$\sum_{i=1}^{d} d^2_{K_{ii}}(u, v) \leq \tilde{N}_K^2 |u - v|^2 \tag{B.9}$$

for some constant $\tilde{N}_K$. Consider the case where K is an explicit kernel. That is to say there exists a finite (p) dimensional feature space and a mapping $\Phi : \mathbb{R}^d \to \mathbb{R}^{p\times d}$ for which:

$$K(u,v) = \Phi(u)^T \Phi(v) \tag{B.10}$$

Note that the Fourier random features used in our experiments fall in this category.

Lemma 4.

$$\sum_{i=1}^{d} d^2_{K_{ii}}(u,v) = \|\Phi(u) - \Phi(v)\|^2_{\mathcal{F}}. \tag{B.11}$$

Where $\mathcal{F}$ is the Frobenious norm.

Proof.

$$\sum_{i=1}^{d} \left(k_{i,i}(u,u) - 2k_{i,i}(u,v) + k_{i,i}(v,v)\right) = \sum_{i=1}^{d} e_i^T \Phi(u)^T \Phi(u) e_i - 2e_i^T \Phi^T(u)\Phi(v)e_i + e_i^T \Phi(v)^T \Phi(v) e_i \tag{B.12}$$

$$= \sum_{i=1}^{d} e_i^T \left\{\Phi(u)^T \Phi(u) - \Phi(u)^T \Phi(v) - \Phi(v)^T \Phi(u) + \Phi(v)^T \Phi(v)\right\} e_i \tag{B.13}$$

$$= \sum_{i=1}^{d} e_i^T \left(\Phi(u) - \Phi(v)\right)^T \left(\Phi(u) - \Phi(v)\right) e_i \tag{B.14}$$

$$= Trace\left(\left(\Phi(u) - \Phi(v)\right)^T \left(\Phi(u) - \Phi(v)\right)\right) \tag{B.15}$$

$$= \|\Phi(u) - \Phi(v)\|^2_{\mathcal{F}}. \tag{B.16}$$

Therefore, for explicit kernels, we conclude that the condition of Lemma 1 is equivalent to the condition that the features are Lipschitz continuous with respect to the Frobenious norm.

B.3. Examples of kernels which satisfy the assumptions of Lemma 1

Let us notate

$$P(u,v) = K_1(u,u) + K_1(v,v) - 2K_1(u,v) \tag{B.17}$$

1. The linear kernel

$$K_1(u,v) = (u^T A v) \tag{B.18}$$

where A is a psd matrix. Indeed,

$$P(u,v) = (u-v)^T A(u-v) \leq ||u-v||^2 \sup_{1\leq i\leq d} \lambda_i \tag{B.19}$$

where λ_i are the eigenvalues of A using the Rayleigh quotient property.

2. The Gaussian kernel:

$$K_1(u,v) = \exp\left(-\frac{1}{2}((u-v)^T A(u-v))\right) \tag{B.20}$$

where A is a psd matrix. Indeed,

$$P(u,v) = 2 - 2\exp\left(-\frac{1}{2}((u-v)^T A(u-v))\right) \leq 2(u-v)^T A(u-v) \leq 2||u-v||^2 \sup_{1\leq I\leq d} \lambda_i \tag{B.21}$$

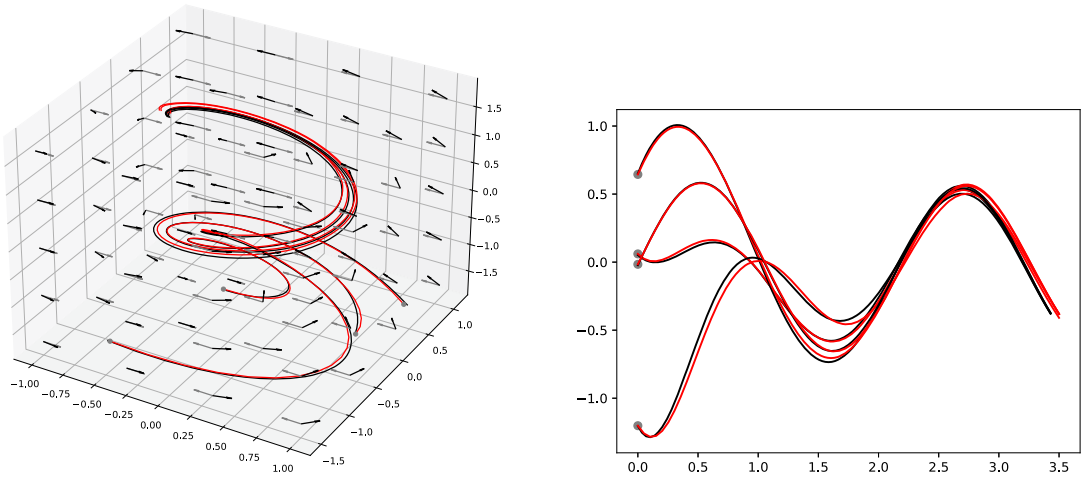
where λ_i are the eigenvalues of A and the first inequality comes from the basic inequality $e^x \geq 1 + x$

3. The rational quadratic kernel:

$$K_1(x,y) = \frac{||x-y||^2}{||x-y||^2 + \theta}, \theta > 0 \tag{B.22}$$

Note that in this case,

$$P(u,v) \leq \frac{1}{\theta}||u-v||^2 \tag{B.23}$$



(a) 3-D Non-autonomous (b) 1-D Solution curves

Fig. C.5. (a): Plot of the 2D system where the z-axis is time. Black arrows: true vector field. Grey arrows: estimated vector field. Black curves: true trajectories. Red curves: estimated trajectories. (b): Grey points: initial conditions. Black curves: true trajectories. Red curves: estimated trajectories.

4. The sinc kernel

$$K_1(u,v) = \prod_{i=1}^{d} \frac{sin(||u_i - v_i||)}{||u_i - v_i||} \tag{B.24}$$

We use the fact that K_1 is a translation invariant kernel with associated density $q(y) = \prod_{i=1}^{d} q_1(y_i)$ with

$$q_1(z) = \frac{1}{2} \text{ for } -1 \le z \le 1 \tag{B.25}$$

5. The Matérn kernel with $p > 3/2$. This kernel is translation invariant with associated density $q(y) = \prod_{i=1}^{d} q_1(y_i)$ with

$$q_1(z) = \frac{1}{(1+x^2)^p} \tag{B.26}$$

and

$$E[X^2] < \infty, X \sim q_1 \tag{B.27}$$

Appendix C. An example of a non-autonomous system

We provide in this appendix a toy example of a non-autonomous system, namely the harmonic oscillator with sinusoidal input force

$$\ddot{y} + 0.001\dot{y} + 10000y = cos(t) \tag{C.1}$$

The kernel is an explicit Fourier random feature kernel with $p = 200$ random features as well as a constant term, where time was included as input together with the spatial variables. Each feature was centered and standardized using the training set only for computing the mean and standard deviation. The functions in the corresponding RKHS are then

$$f([x_1,x_2,t]) = \begin{bmatrix} \sum_{i=1}^{p} \alpha_i \cos([z_{1i}, z_{2i}, z_{3,i}] \cdot [x_1,x_2,t]) + \beta_i \sin([z_{1i}, z_{2i}, z_{3,i}] \cdot [x_1,x_2,t]) + \omega_1 \\ \sum_{i=1}^{p} \gamma_i \cos([z_{1i}, z_{2i}, z_{3,i}] \cdot [x_1,x_2,t]) + \delta_i \sin([z_{1i}, z_{2i}, z_{3,i}] \cdot [x_1,x_2,t]) + \omega_2 \end{bmatrix} \tag{C.2}$$

Where the z variables are iid sampled from a standard Normal (or Gaussian) distribution and the parameters $\{\alpha_i, \beta_i, \gamma_i, \delta_i\}, i = 1 \dots p$ along with $\{\omega_j\}, j = 1, 2$ are learned from the training set. Fig. C.5 illustrates the output ODE-RKHS algorithm for this system.

References

[1] M.W. Hirsch, S. Smale, R.L. Devaney, Differential Equations, Dynamical Systems, and an Introduction to Chaos, Academic Press, 2012.
[2] J.H. Manton, P.-O. Amblard, et al., A primer on reproducing kernel hilbert spaces, Found. Trends Signal Process. 8 (2015) 1–126.
[3] F. Dondelinger, D. Husmeier, S. Rogers, M. Filippone, Ode parameter inference using adaptive gradient matching with gaussian processes, in: Artificial Intelligence and Statistics, PMLR, 2013, pp. 216–228.
[4] S.L. Brunton, J.L. Proctor, J.N. Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems, Proc. Natl. Acad. Sci. 113 (2016) 3932–3937.

[5] M. Niu, S. Rogers, M. Filippone, D. Husmeier, Fast parameter inference in nonlinear dynamical systems using iterative gradient matching, in: International Conference on Machine Learning, PMLR, 2016, pp. 1699–1707.
[6] P. Hu, W. Yang, Y. Zhu, L. Hong, Revealing hidden dynamics from time-series data by odenet, arXiv preprint arXiv:2005.04849, 2020.
[7] T. Qin, K. Wu, D. Xiu, Data driven governing equations approximation using deep neural networks, J. Comput. Phys. 395 (2019) 620–635.
[8] R.T. Chen, Y. Rubanova, J. Bettencourt, D. Duvenaud, Neural ordinary differential equations, arXiv preprint arXiv:1806.07366, 2018.
[9] B.O. Koopman, Hamiltonian systems and transformation in hilbert space, Proc. Natl. Acad. Sci. USA 17 (1931) 315.
[10] P.J. Schmid, Dynamic mode decomposition of numerical and experimental data, J. Fluid Mech. 656 (2010) 5–28.
[11] X. Dai, L. Li, Kernel ordinary differential equations, J. Am. Stat. Assoc. 117 (2022) 1711–1725.
[12] M. Heinonen, C. Yildiz, H. Mannerström, J. Intosalmi, H. Lähdesmäki, Learning unknown ode models with gaussian processes, in: International Conference on Machine Learning, PMLR, 2018, pp. 1959–1968.
[13] M. Kanagawa, P. Hennig, D. Sejdinovic, B.K. Sriperumbudur, Gaussian processes and kernel methods: a review on connections and equivalences, arXiv preprint arXiv:1807.02582, 2018.
[14] T. Hofmann, B. Schölkopf, A.J. Smola, Kernel methods in machine learning, Ann. Stat. (2008) 1171–1220.
[15] M.A. Alvarez, L. Rosasco, N.D. Lawrence, Kernels for vector-valued functions: a review, arXiv preprint arXiv:1106.6251, 2011.
[16] G.F. Simmons, Differential Equations with Applications and Historical Notes, Third edition, CRC Press, 2016, Chapter 13, Theorem B.
[17] M. Raissi, P. Perdikaris, G. Karniadakis, Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys. 378 (2019) 686–707.
[18] T. Dao, C. De Sa, C. Ré, Gaussian quadrature for kernel features, Adv. Neural Inf. Process. Syst. 30 (2017) 6109.
[19] E. Kiliç, P. Stanica, The inverse of banded matrices, J. Comput. Appl. Math. 237 (2013) 126–135.
[20] R. Vershynin, High-Dimensional Probability: An Introduction with Applications in Data Science, vol. 47, Cambridge University Press, 2018.
[21] J. Quinonero-Candela, C.E. Rasmussen, A unifying view of sparse approximate gaussian process regression, J. Mach. Learn. Res. 6 (2005) 1939–1959.
[22] P. Kokotovic, J. Heller, Direct and adjoint sensitivity equations for parameter optimization, IEEE Trans. Autom. Control 12 (1967) 609–610.
[23] P. Zheng, T. Askham, S.L. Brunton, J.N. Kutz, A.Y. Aravkin, A unified framework for sparse relaxed regularized regression: Sr3, IEEE Access 7 (2018) 1404–1423.
[24] B. de Silva, K. Champion, M. Quade, J.-C. Loiseau, J. Kutz, S. Brunton, Pysindy: a python package for the sparse identification of nonlinear dynamical systems from data, J. Open Sour. Softw. 5 (2020) 2104.
[25] E. Lew, A. Hekal, K. Potomkin, N. Kochdumper, B. Hencey, S. Bak, S. Bogomolov, Autokoopman: a toolbox for automated system identification via koopman operator linearization, in: International Symposium on Automated Technology for Verification and Analysis, Springer, 2023, pp. 237–250.
[26] M.O. Williams, I.G. Kevrekidis, C.W. Rowley, A data–driven approximation of the koopman operator: extending dynamic mode decomposition, J. Nonlinear Sci. 25 (2015) 1307–1346.
[27] E. Yeung, S. Kundu, N. Hodas, Learning deep neural network representations for koopman operators of nonlinear dynamical systems, in: 2019 American Control Conference (ACC), IEEE, 2019, pp. 4832–4839.
[28] A.M. DeGennaro, N.M. Urban, Scalable extended dynamic mode decomposition using random kernel approximation, SIAM J. Sci. Comput. 41 (2019) A1482–A1499.
[29] Z. Zhao, D. Giannakis, Analog forecasting with dynamics-adapted kernels, Nonlinearity 29 (2016) 2888.
[30] D. Burov, D. Giannakis, K. Manohar, A. Stuart, Kernel analog forecasting: multiscale test problems, Multiscale Model. Simul. 19 (2021) 1011–1040.
[31] H. Schaeffer, G. Tran, R. Ward, L. Zhang, Extracting structured dynamical systems using sparse optimization with very few samples, Multiscale Model. Simul. 18 (2020) 1435–1461.
[32] E. Lorenz, Predictability: a problem partly solved, Ph.D. thesis, Shinfield Park, Reading, 1995.
[33] M.P. Murphy, H. LeVine III, Alzheimer's disease and the amyloid-β peptide, J. Alzheimer's Dis. 19 (2010) 311–323.
[34] P. Vernhet, M. Bilgel, S. Durrleman, S.M. Resnick, S.C. Johnson, B.M. Jedynak, Modeling the early accumulation of amyloid using differential equations in wrap and blsa: neuroimaging/optimal neuroimaging measures for early detection, Alzheimer's Dement. 16 (2020) e039536.
[35] J.L. Doob, Regularity properties of certain families of chance variables, Trans. Am. Math. Soc. 47 (1940) 455–486.
[36] K.E. Atkinson, An Introduction to Numerical Analysis, John Wiley & Sons, 2008.