Overcoming error-in-variable or stiffness problem in SINDy-like data-driven model discovery

LLOYD FUNG, Imperial College London, UK

Despite the recent proliferation of machine learning methods like SINDy that promise automatic discovery of governing equations from time-series data, there remain significant challenges to discovering models from noisy datasets. One reason is that the linear regression underlying these methods assumes that all noise resides in the training target (the regressand), which is the time derivative, whereas the measurement noise is in the states (the regressors). Recent methods like modified-SINDy and DySMHO address this error-in-variable problem by leveraging information from the model's temporal evolution, but they are also imposing the equation as a hard constraint, which effectively assumes no error in the regressand. Without relaxation, this hard constraint prevents assimilation of data longer than Lyapunov time. Instead, the fulfilment of the model equation should be treated as a soft constraint to account for the small yet critical error introduced by numerical truncation. The uncertainties in both the regressor and the regressand invite the use of orthogonal distance regression (ODR). By incorporating ODR with the Bayesian framework for model selection, we introduce a novel method for model discovery, termed ODR-BINDy, and assess its performance against current SINDy variants using the Lorenz63, Rössler, and Van Der Pol systems as case studies. Our findings indicate that ODR-BINDy consistently outperforms all existing methods in recovering the correct model from sparse and noisy datasets. For instance, our ODR-BINDy method reliably recovers the Lorenz63 equation from data with noise contamination levels of up to 30%.

CCS Concepts: • **Applied computing** → *Mathematics and statistics*; • **Mathematics of computing** → **Time series analysis**; *Convex optimization*; **Ordinary differential equations**; *Stochastic differential equations*; **Bayesian computation**; **Max marginal computation**; **Computing most probable explanation**; *Statistical software*; Quasiconvex programming and unimodality; • **Computing methodologies** → **Supervised learning by regression**; **Data assimilation**.

Additional Key Words and Phrases: SINDy, model discovery, error-in-variable, data assimilation, chaos, Bayesian evidence

ACM Reference Format:
Lloyd Fung. 2026. Overcoming error-in-variable or stiffness problem in SINDy-like data-driven model discovery. *ACM Trans. AI Sci.* 1, 1, Article 1 (June 2026), 27 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Despite the recent proliferation of black-box machine learning techniques such as Neural Networks, symbolic or equation-based models remain critical in many scientific disciplines owing to their interpretability and parsimonious nature. The sparse identification of nonlinear dynamics (SINDy) technique [6] has become one of the most popular techniques to learn the governing dynamical equation from observed time-series data of the full system state. Starting from a library of candidate functions that presumably contain the terms required to describe the system, SINDy's algorithm regresses the time derivative estimated from the time-series data of the states against the nonlinear

Author's Contact Information: Lloyd Fung, lloyd.fung@imperial.ac.uk, Imperial College London, London, UK.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2026 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM 3066-4438/2026/6-ART1
https://doi.org/XXXXXXX.XXXXXXX

functions of its state while promoting sparsity in the resulting learnt parameters. The resulting parsimonious model was shown to share the same sparsity pattern as the original model, effectively demonstrating that the SINDy algorithm can recover governing equations from data given an appropriate library of nonlinear functions and coordinates for the observed dynamics. Owing to its simplicity and speed, since its inception, the method has been applied to many applications across disciplines. It has also attracted and inspired many improvements, extensions and variants, ranging from alternative sparsity-promoting regression algorithms (SR3) [29], extensions to finding PDEs [26], extensions to libraries of rational functions [14], improvements to noise robustness by ensembling [8] to the incorporation of active learning strategies [9].

Despite its utility, SINDy's sparsity promotion process is inherently empirical. The underlying Sequential Threshold Least Squares (STLS) algorithm promote sparsity by removing terms with coefficients smaller than a certain threshold after each successive least squares regression. This process, while effective, prevents the algorithm from learning potentially multiscale dynamics, as the threshold is fixed across terms. Recent works [3, 9, 11, 22, 28] have shown that by adopting a Bayesian perspective on the sparsity-promoting process, one can not only improve noise robustness in recovering the correct model but also put the sparsity promotion on a more rigorous foundation. The key advantage of the Bayesian interpretation of SINDy is that the principle of Occam's razor that underlies the argument for a parsimonious model naturally arises when one maximizes the Bayesian evidence or marginalised likelihood [20]. Another approach to improve the robustness of SINDy is to use it in conjunction with ensembling [8], which, as shown by recent work [10], can also be interpreted as a Bayesian approach.

Although the Bayesian reformulation can somewhat improve noise robustness, as demonstrated empirically by recent works [8, 9, 22], large noise remains challenging to the SINDy framework and many of its variants. This challenge of noise is twofold. First, given some time-series data $\hat{\mathbf{X}}$, the SINDy framework relies on a good approximation of the time derivatives of time-series data $\dot{\mathbf{X}}$, in which the dot $\dot{_}$ denotes the time derivative. Classical methods to approximate time derivatives, like finite difference, tend to amplify the measurement noise in time-series data. To alleviate this noise, Reinbold et al. [24] and Messenger and Bortz [21] indepedently introduced the weak formulation. It is conceptually similar to applying a finite element method in the time domain, as the weak form projects the time series data onto some local trial function space. Irrespective of how the time derivatives are approximated, whether via finite difference, weak formulation, or other collocation methods, they can all be written as a linear transformation of the time series. Therefore, the propagation of noise is tractable and easily captured by the Bayesian method. By taking into account the noise variance amplification by the time-derivative approximation, as demonstrated by Fung et al. [9], a Bayesian formulation can alleviate this noise amplification issue. Meanwhile, the second challenge posed by large noise is difficult to address even with the aforementioned Bayesian extension. The challenge arises from the error-in-variable noise, i.e. noise in $\mathbf{X}$. As this noise is propagated through the nonlinear dictionary $\boldsymbol{\Theta}(\mathbf{X})$, it gets amplified and distorted by highly nonlinear functions. Previous works [9] accounted for this noise contamination of the dictionary to some extent by propagating the noise through a local linear approximation to the dictionary. However, the propagation only works if the noise magnitude is small enough for the approximation. For large noise, propagation through a nonlinear term will result in a highly non-Gaussian or even multimodal distribution that evades simple regression techniques.

Nonetheless, learning an equation from signals with a high noise-to-signal ratio is not completely impossible. The key insight is that information regarding the noise-contaminated state at a certain time is also contained in the neighbouring data points in time. The weak formulation is one way to extract this information, as it effectively projects the neighbourhood of a certain time onto a more

plausible (and often smoother) subspace of temporal basis functions. In a follow-up paper, Bortz et al. [5] argued that the weak formulation is an effective way to overcome the error-in-variable problem. Indeed, their results show that this is a plausible and fast method. However, this approach inevitably introduces spectral bias, as it effectively assumes the trajectory follows a certain shape in the local time domain, which described by a linear combination of the trial basis functions used. Indeed, both Messenger and Bortz [21] and Fung et al. [9] have discussed the implication of the weak formulation as a low-pass filter. Hence, the use of weak formulation can be interpreted as a form of inductive bias that regularises the learning problem. It encodes the user's prior knowledge of the frequency of the true signal. If the data is sampled at a much higher frequency than the signal, the weak formulation is an effective way to filter the noise away from the signal.

A less biased approach is to use the model itself to extract information from neighbouring points. After all, the model, in contrast to the weak form trial basis, is the true statement that describes the relationship between neighbouring points. The key task in this approach is, therefore, to leverage the learnt model to estimate the original states $\mathbf{X}$ from information provided by neighbouring data points. However, the model is not known *a priori*. Hence, the challenge is to simultaneously denoise the data and learn the equation in a single optimization. Several recently proposed methods, including modified-SINDy [13], DAHSI [25], DySMHO [17] and SIDDs [12], have attempted this approach with various nonlinear optimization and modeling strategies. Modified-SINDy [13] put the numerical integration of the ODE inside an optimization that optimizes the model parameter and promotes sparsity while minimizing the discrepancy between the trajectorial data and the integrated path. DAHSI [25], DySMHO [17], and SIDDs [12] used a similar strategy but discretised the equation by some collocation or finite difference method first before optimization, thereby preventing the use of expensive ODE solvers within the optimization. These works have claimed improvements in noise robustness and accuracy in the parameter estimate compared to previous linear regression-based methods [8].

However, with the exception of DAHSI [25], a common feature in the aforementioned works [12, 13, 17] is that the differential equation acts as a hard constraint that needs to be satisfied exactly during optimization. For modified-SINDy [13], this enforcement comes in the form of integrating the ODE with classic solvers. For DySMHO [17] and SIDDs [12], both of which employ a discretise-then-optimize approach, the enforcement of the equation comes in the form of a constrained optimization, in which the denoised trajectories are constrained to follow a set of parameterised equations, the parameter and form of which are to be "learnt" via sparsity-promoting optimization. The "hard" constraint in these optimizations creates stiffness in the optimization that prevents assimilation of data longer than the Lyapunov timescale when the system is chaotic. To relax this stiffness, all of the aforementioned works have implicitly applied some form of relaxation empirically, which we shall more explicitly point out in § 2.1.

In this work, we introduce ODR-BINDy, which also exploits the noise robustness of the denoising strategy. In many ways, ODR-BINDy is similar to modified-SINDy, DySMHO, DAHSI and SIDDs in its ability to denoise and learn simultaneously. Much like DySMHO, DAHSI, and SIDDs, ODR-BINDY also employs a discretize-then-optimize strategy. However, there are a few key differences where ODR-BINDy stands out. 1. ODR-BINDy uses Bayesian evidence to select models, which has been shown to be more rigorous and robust. 2. ODR-BINDy employs a more traditional Newton's method to optimize the underlying nonlinear least square problem, which, as we shall demonstrate later, converges much quicker with a good initial guess, which one can find by ensemble linear regression. 3. Most importantly, ODR-BINDy relaxes the fulfilment of the differential equation by imposing it as a soft constraint, making it more robust against discretisation error or stochastic noise.

Before moving on, we should stress that this work focuses purely on improving the noise robustness of the SINDy/BINDy framework. Thus, within the scope of this work, we will presume that the observed representation (i.e. the data) of the underlying dynamical system is the one where sparse representation is possible. Under this presumption, the general discussion on the choice of representation, the design of the initial dictionary and the nature of sparsity promotion will be categorised as outside the scope of the current work. Readers are encouraged to consult previous literature on the matter of the representation [7] and design choice of the dictionary [6].

2 Formulation

We start by introducing the formulation of SINDy or a typical SINDy variant. Firstly, it assumes that the data is driven by an underlying (autonomous) dynamical system that can be written in the form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t), \Xi) = \boldsymbol{\Theta}(\mathbf{x}(t))\Xi, \tag{1}$$

where $\mathbf{x}(t) = [x_1(t)\ x_2(t)\ \dots\ x_j(t)\ \dots\ x_D(t)] \in \mathbb{R}^D$ are the states of a D-dimensional dynamical system, $\boldsymbol{\Theta}(\mathbf{x}(t)) \in \mathbb{R}^M$ a dictionary of M nonlinear functions of the states, and $\Xi \in \mathbb{R}^{M\times D}$ the parameters that parameterise the dynamical system. SINDy then assumes that a parsimonious description of the function $\mathbf{f}$ would imply sparsity in the parameter matrix $\Xi \in \mathbb{R}^{M\times D}$, given appropriate choices for the coordinates and dictionary of the states. Therefore, by approximating $\dot{\mathbf{x}}$ with the appropriate method and regressing it against $\boldsymbol{\Theta}(\mathbf{x}(t))$ while promoting the correct sparsity, one can effectively learn the functional form of $\mathbf{f}$. It should be noted that $\mathbf{f}$ does not necessarily always have the form described in (1), i.e., linearly summed nonlinear functions of $\mathbf{x}$. In particular, DySMHO [17] and DAHSI [25] have shown how the framework can be extended beyond (1).

In this work, we will assume the data comes from **full and unbiased observation of all the states**, meaning that we can directly measure all states $\hat{\mathbf{x}}(t_i)$ at time $t = t_1, t_2, \dots t_i \dots t_{\tilde{N}}$, with some unbiased measurement noise $\boldsymbol{\epsilon}_x$, such that

$$\hat{X}_{ij} = \hat{x}_j(t_i) = x_j(t_i) + \epsilon_{x,ij} = X_{ij} + \epsilon_{x,ij}. \tag{2}$$

For simplicity, we assume the sampling is done at a fixed time interval, although this is not necessarily a restriction for the methodology. However, SINDy-like methodologies, as well as this work, necessitate that the sampling be frequent enough. For example, in the context of chaotic systems, the sampling should be more frequent than the inverse of Lyapunov time. We will also assume that the noise $\boldsymbol{\epsilon}_x \in \mathbb{R}^{\tilde{N}\times D}$ consists of $\tilde{N} \times D$ i.i.d. random variables with Gaussian distributions, with zero means (i.e. unbiased) and variance $\boldsymbol{\sigma}_x^2 \in \mathbb{R}^{\tilde{N}\times D}$. Extensions to other noise models are possible by substituting the loss function in the next step with the corresponding energy function (negative log of the probability distribution representing the noise model). For simplicity, here we assume Gaussian noises.

Given the discrete sampling, the dictionary $\boldsymbol{\Theta}(\mathbf{x}(t))$ is rewritten as a matrix $\boldsymbol{\Theta}(\mathbf{X}) \in \mathbb{R}^{\tilde{N}\times M}$, where each row represents each time point, and each column represents each nonlinear function. For example, if the dictionary is assumed to be polynomials of the states $\mathbf{x}(t) = [x_1(t), x_2(t), x_3(t)]$ of a three dimensional system (i.e. $D = 3$), we can write down the dictionary as

$$\boldsymbol{\Theta}(\mathbf{X}) = \begin{bmatrix} 1 & X_{11} & X_{12} & X_{13} & X_{11}^2 & X_{11}X_{12} & X_{11}X_{13} & X_{12}^2 & X_{12}X_{13} & X_{13}^2 & \dots \\ 1 & X_{21} & X_{22} & X_{23} & X_{21}^2 & X_{21}X_{22} & X_{21}X_{23} & X_{22}^2 & X_{22}X_{23} & X_{23}^2 & \dots \\ 1 & X_{31} & X_{32} & X_{33} & X_{31}^2 & X_{31}X_{32} & X_{31}X_{33} & X_{32}^2 & X_{32}X_{33} & X_{33}^2 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

up to some order that also determines M. To approximate the time derivative from the time-series data $\mathbf{X}$, we assume there exist some linear operators in the form of matrices $L_I \in \mathbb{R}^{N\times\tilde{N}}$ and

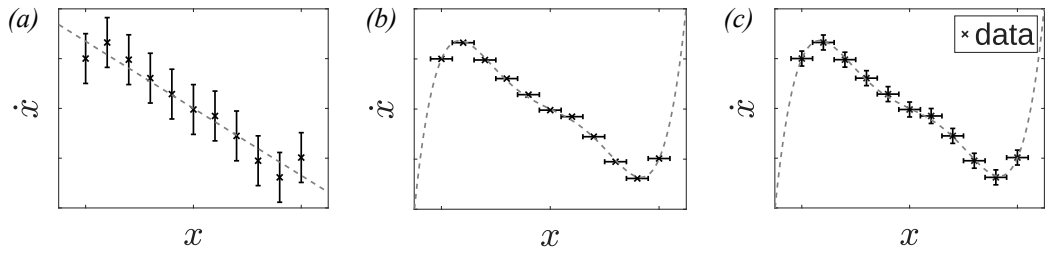


Fig. 1. Visualisation of noise type by the direction of errorbars. (a) Linear regression-based methods, like SINDy, assumed that the noise is in the $\dot{\mathbf{x}}$ direction only. (b) Nonlinear optimization-based methods, like modified-SINDy, DySMHO and SIDDs, uses hard constrain to satisfy the equation, equivalent to assuming no noise in the $\dot{\mathbf{x}}$ direction. Hence, noise is in the $\mathbf{x}$ direction only. (c) Soft-constrained nonlinear optimization-based methods, like DAHSI and ODR-BINDy, assumed noise in both the $\mathbf{x}$ and $\dot{\mathbf{x}}$ direction. Regression with noise in the regressors is also called error-in-variable analysis. One technique to perform this regression is the orthogonal distance regression (ODR) [4].

$L_{\partial t} \in \mathbb{R}^{N \times \tilde{N}}$ such that

$$L_I \dot{\mathbf{X}}_i \approx L_{\partial t} \mathbf{X}, \quad \text{and therefore} \quad L_{\partial t} \mathbf{X} \approx L_I \mathbf{f}(\mathbf{X}) \tag{3}$$

Both finite difference and weak formulation, as well as other collocation-based methods that discretise the time domain, can be written in this form. Moreover, L_I and $L_{\partial t}$ are usually sparse or banded, accelerating the computation in later steps. A detailed discussion on these operators can be found in [9].

Here, ODR-BINDy, along with the aforementioned works [12, 13, 17, 25], departs from classical linear regression-based SINDy variants. Classical SINDy-like approaches presumed that all noise resides in the regressand, specifically on the left-hand side of the second equation of (3). In particular, [9] had spent considerable effort approximating how noise in $\mathbf{X}$ propagates through the nonlinear dictionary $\boldsymbol{\Theta}(\mathbf{X})$ such that one can effectively treat the noise as solely present the regressand and not in the regressors. Fig. 1(a) illustrates this approach in a simple 1D case, in which the noise is expressed as error bars in the $\dot{\mathbf{x}}$ direction. The advantage of this approach lies in the assumption that the effective noise exists just in the regressand, allowing us to utilize basic linear regression to reduce the least squares. The drawback, however, is that the effective noise may not conform to a Gaussian distribution, as presumed by the least square method. This is especially true when large noise is propagating through highly nonlinear functions, thereby limiting the method's robustness against large noise.

However, if we stay true to the noise model described in (2), the noise should have been in the regressand instead of the regressor. Visually, we can represent this regression as a minimization in the regressor error as shown in fig. 1(b), which is in contrast to the minimization of regressand error by linear regression. If we also assume a Gaussian prior on the parameter, the justification of which will be discussed in § 2.2 (also see [9]), then the loss function arising from the log-posterior distribution should be

$$\mathcal{L}(\mathbf{X}, \Xi) = \frac{1}{2\sigma_x^2} ||\hat{\mathbf{X}} - \mathbf{X}(\Xi)||_F^2 + \frac{1}{2\sigma_p^2} ||\Xi||_F^2, \tag{4}$$

where $\mathbf{X}(\Xi)$ is constrained by the discretised differential equation

$$L_{\partial t} \mathbf{X} - L_I \boldsymbol{\Theta}(\mathbf{X}) \Xi = 0. \tag{5}$$

In this case, we can avoid the inconveniences of approximating noise propagation through the nonlinear functions in $\boldsymbol{\Theta}(\mathbf{X})$. This approach is analogous to how linear regression is extended

to nonlinear regression in a typical error-in-variables (EIV) analysis, and form the basis for the standard way to perform parameter estimation by data assimilation [1, 18]. Indeed, minimizing (4) and (5) is akin to the strong constraint four-dimensional variational (4D-Var) assimilation technique [1, chap. 2.4.3.2]. DySMHO, SIDDs and modified-SINDy are also solving this optimization problem (or its equivalence). We also note that the loss in (4) can be easily modified to account for noise models other than Gaussian. Although this nonlinear optimization has no guarantee for convexity, it is not a major constraint provided that we have a good initial guess. This initial guess can be from the data itself or some pre-processed form of it that has already smoothed some of the data (see § 2.3).

2.1 The importance of stochastic relaxation

Now, besides a slightly different prior and sparsity promotion methodology, (4) and (5) are effectively the optimization problem underpins DySMHO and SIDDs. Modified-SINDy is also solving an equivalent problem, but it enforced the constraint by evaluating the ODE with a solver rather than use collocation in the time domain. Here, our ODR-BINDy approach and DAHSI differ from DySMHO, modified-SINDy, and SIDDs: **Rather than strictly enforcing (5), we define the residue to (5) as a small noise**

$$\boldsymbol{\eta} = L_{\partial t}\mathbf{X} - L_I\boldsymbol{\Theta}(\mathbf{X})\Xi.$$

The rationale for this noise $\boldsymbol{\eta}$ is as follows. Real-life data seldom adhere precisely to the dynamics outlined by (1). Instead, there is always small stochasticity that results in the state deviating somewhat from the anticipated dynamics described by (1). Even with synthetic data where stochastic noise is ostensibly absent, numerically solving (1) will inevitably introduce truncation error. The collocation method that gives rise to $L_{\partial t}$ and L_I also introduces truncation error. In other words, (5) can never be satisfied exactly (except when the identical discretisation scheme is employed for both data generation and assimilation, which is rarely the case in practice), and the noise $\boldsymbol{\eta}$ represents any discrepancy between the learnt model and the dynamic that generated the data.

Now, in systems without any positive Lyapunov exponent, a small $\boldsymbol{\eta}$ will not prevent methods like DySMHO, SIDDs and modified-SINDy from enforcing (5) to the best accuracy possible when solving the inverse problem through optimizing (4) . However, for chaotic systems or any system with a positive Lyapunov exponent, the small error $\boldsymbol{\eta}$ will accumulate in time, preventing $\mathbf{X}$ from following the data $\hat{\mathbf{X}}$ for a time longer than the Lyapunov time scale. In other words, if (5) is enforced exactly without relaxation, i.e., as a "hard" constraint, a small mismatch between how the data is generated (1) and how the data is modelled (5) or slight stochasticity in the real-life data will prevent $\mathbf{X}$ from following the data $\hat{\mathbf{X}}$ longer than the Lyapunov time scale.

Fig. 2 illustrates this difficult in a prototypical problem of parameter estimation in Lorenz63, whereby data is assimilated to estimate the parameters of know equations of Lorenz63 by optimizing (4) under the constrain (5). It is evident that without relaxing the hard constraint, the model cannot fully adhere to the data.

We should note that this is not a novel discovery. The challenge of assimilating data beyond the Lyapunov time scale is well established [1, 2, 23]. However, in the context of model discovery, DAHSI is the sole notable study known to us that explicitly incorporates the model error $\boldsymbol{\eta}$ as a soft constraint rather than a hard constraint. Instead, most studies [12, 13, 17] address the issue through empirical relaxation during training. For instance, SIDDs [12] implicitly relaxed the constraint (5) during the training, enforcing (5) only at termination. Modified-SINDy empirically penalises losses that are not temporally local through the ω prefactor in eq. (11-12) of [13]. Meanwhile, DySMHO [17] limited the time horizon to be smaller than the Lyapunov time scale. While these empirical relaxations somewhat attenuate the issue, they function more as a hotfix than a theoretically

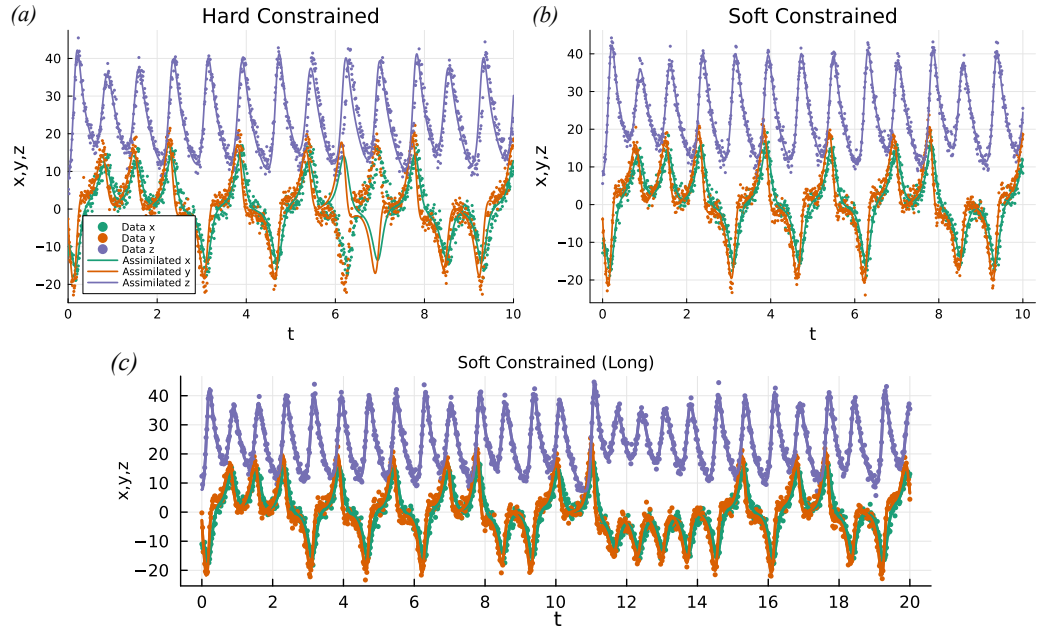


Fig. 2. Data assimilation of Lorenz63 with (a) optimization upon hard constrained ODE solver and (b) with soft constrained loss function. Data are sampled at $\Delta t = 0.01$ with noise $\sigma_x = 1.6$. Notice how the assimilated trajectory diverge at $t \sim 7$ in (a). This is despite our best effort to optimize the parameter and the initial condition. This is because the constraint (5) do not account for the any numerical error that will lead to divergence longer than Lyapunov time scale. In contrast, the soft constrain (6) use in (b) relax the problem such that the assimilated data can always follow the data in the long time limit, even beyond Lyapunov time scale. However, the data must be sampled more frequently than Lyapunov time. (c) Assimilation of a longer trajectory using the same soft constraint as (b) to demonstrate that the method indeed works for any arbitrarily long series of data.

informed relaxation and may hinder the method's capacity to maximize learning from the given data.

In this study, we propose that rather than relaxing the issue by other means, we should treat the error $\boldsymbol{\eta}$ arising from numerical truncation or small stochasticity properly as part of the loss. Here, we empirically model $\boldsymbol{\eta}$ as an independent and identically distributed noise $\boldsymbol{\eta} \sim \mathcal{N}(0, \sigma_{\partial t}^2 \mathbf{I})$, characterised by a Gaussian distribution with variance $\sigma_{\partial t}^2$, which quantifies the magnitude of such error. The choice of a Gaussian distribution is primarily for mathematical convenience, as it allows us to derive a simple least square loss function and exploit the orthogonal distance [4] to easily compute the Laplace approximation in the parameter space for Bayesian model selection (see §A.2.1). We should stress that modeling $\boldsymbol{\eta}$ as a zero-mean Gaussian random variable is likely a gross oversimplification, as the model error arising from numerical truncation rarely has no bias, nor does it conform to a Gaussian distribution. It is also likely that the magnitude of the error depends on the states themselves (i.e., heteroscedasticity). Numerical error at one time point also likely correlates with other neighboring time points, so the assumption of independence between time points is also an oversimplification.

However, accurately modeling the aforementioned complexity in $\boldsymbol{\eta}$ may incur significant computational cost with minimal enhancement in achieving the goal of selecting the correct model.

For instance, addressing the temporal correlation necessitates the creation of a covariance matrix of dimensions ND, which is likely to impose considerable computational burden. To address heteroscedasticity, the noise variance must be modeled as a function of the states, necessitating additional parameters to be optimized and an additional model for the noise. Given the primary factors influencing model selection are whether $\mathbf{X}$ from the model can follow $\hat{\mathbf{X}}$ in the long term and whether the approximation to the Bayesian evidence is adequate for comparison between models, the simple Gaussian noise model is likely sufficient to fulfill the goal of this work. Therefore, **while this simplification may not be ideal, it is a reasonable first step towards relaxing the hard constraints in the context of model discovery.**

Moreover, it can be argued that in real-life applications, the noise $\boldsymbol{\eta}$ is the result of a combination of truncation error and real-world stochasticity in the system. Therefore, the precise noise model for $\boldsymbol{\eta}$ will be problem-specific. Nonetheless, by the virtue of the central limit theorem, a Gaussian noise model is likely a reasonable approximation for the combined effect of truncation error and real-world stochasticity. In future extensions, the loss term representing $\boldsymbol{\eta}$, i.e.,the model loss, may be substituted with alternative noise models to better capture the true nature of $\boldsymbol{\eta}$. However, it should be noted that more complex noise models may come at the cost of computational complexity. With the injection of this noise, the model equation becomes analogous to a random ODE or SDE; thus, we shall designate this technique as **stochastic relaxation**. The phrase also highlights the fact that although the system to be learnt is deterministic, to assimilate data beyond Lyapunov time, one must treat it as if it has some stochasticity in order to overcome the system's sensitivity to small perturbations arising from numerics. While this modification may seem inconsequential at first, its significance cannot be understated. The fact that we allow the learning algorithm to adjust the integration of (5), however small the adjustment is, allows the integrated trajectory from (5) to always follow the data, even beyond Lyapunov timescale. This helps the algorithm maximize the information gain from a dataset by assimilating long trajectories in a single optimization, which is in contrast to the moving horizon strategy of DySMHO and the empirical penalisation of modified-SINDy. In practice, stochastic relaxation reformulate the problem into an unconstrained optimization for the loss

$$\mathcal{L}(\mathbf{X}, \Xi) = \underbrace{\frac{1}{2\sigma_x^2}||\hat{\mathbf{X}} - \mathbf{X}||_F^2}_{\text{Data Loss}} + \underbrace{\frac{1}{2\sigma_{\partial t}^2}||L_{\partial t}\mathbf{X} - L_I\boldsymbol{\Theta}(\mathbf{X})\Xi||_F^2}_{\text{Model Loss}} + \underbrace{\frac{1}{2\sigma_p^2}||\Xi||_F^2}_{\text{Prior}}. \tag{6}$$

in which the first term represents the data loss, arise from minimizing the measurement noise $\boldsymbol{\epsilon}_x$, the second term represents the model loss, arise from minimizing the stochastic noise $\boldsymbol{\eta}$, and the third term represents a Gaussian prior on the coefficients. Each term is scaled by the corresponding noise and prior variance. In other words, we have essentially regarded (5) as a "soft" constraint rather than a "hard" constraint. We designate the first term as data loss, next term as model loss, and the last term as the prior. Each term is scaled by the corresponding noise and prior variance. Visually, fig. 1(c) depicts the noise variances as error bars: the error bar in the $\mathbf{X}$ direction depicts σ_x, and the error bar in the $\mathbf{X}$ direction depicts $\sigma_{\partial t}$. Since the noise is in both the $\mathbf{X}$ direction and $\dot{\mathbf{X}}$ direction, the overall optimization is analogous to Orthogonal Distance Regression (ODR) [4], a specific type of nonlinear least square optimization where the error is minimized in both the regressand and regressor. While the original ODR algorithm is well established and has provable convergence properties [4], it is, in fact, a special case of classical trusted-region Newton's method. Therefore, we will employ general Newton's method to minimize (6).

Here, we note that optimizing (6) is akin to the weak constraint four-dimensional variational (4D-Var) technique in data assimilation [1, chap. 2.4.3.2]. It is also equivalent to the loss function

used in Physics-Informed Neural Network (PINN) [c.f. 16, eq. 3], in which $1/\sigma_x^2$ and $1/\sigma_{\partial t}^2$ acts as pre-factors of the data loss and model loss, respectively. Meanwhile, the prior term is often not explicitly included in PINN but can be easily added as a regulariser.
Notably, the algorithm DAHSI also shares the same loss function. However, our approach to managing the hyperparameters and the philosophy behind the derivation of the loss function is different. In DAHSI's variational annealing method, the scaling pre-factor of the model loss is gradually adjusted throughout the optimization. In contrast, we fix each hyperparameter in accordance to their mathematical origin: each pre-factor scaling the loss functions is the inverse of the corresponding noise and prior variance, fixed according to the user's knowledge or assumptions regarding measurement noise, stochastic noise or truncation error, and the prior. More discussion on how hyperparameters are chosen will be discussed in § 2.4.

2.2 Bayesian model selection

After the loss function is established, the subsequent step is implementing the model selection process. To this end, we utilize the same Bayesian model selection approach as outlined in [9]. First, we calculate, with a suitably chosen Gaussian prior, the Bayesian evidence via Laplace Approximation for each model after the stochastically relaxed optimization for the parameters of each model. We then search for the model (i.e. the combination of terms) that maximizes the evidence using a Greedy search algorithm (see algorithm 1). The resulting model selected by this process is the one that best explains the data while also being parsimonious, as the evidence maximization naturally balances the level of fit to the data and the sparsity in the model. To expedite calculations and guarantee that the parameter variance remains positive definite, we also utilize the Gauss-Newton approximation for the evidence, as this approximation appears to have minimal impact on the model selection process (see § A.5).
The benefits of choosing a model based on Bayesian evidence, as opposed to thresholding or other empirical methods, have been extensively examined in [9]. We succinctly summarise the findings herein. The Bayesian evidence measures the likelihood of the data conditional on the hypothesised model. Assuming that all potential models (i.e., combinations of candidate functions in the library) are equally probable, the evidence measures the extent to which a model explains the data. It also naturally encourage more parsimonious models [19]. Consequently, it serves as a mathematically rigorous metric for comparing different models, i.e. different combinations of terms in the context of sparsity promotion [20]. Identifying the model that optimizes evidence is also referred to as type-II maximum likelihood [27]. Unlike the empirical thresholding procedure of STLS that underpins SINDy and its numerous variants, which often eliminates significant terms with small coefficients, the Bayesian method does not discards coefficients according to their magnitude. Instead, it eliminates coefficients are too sensitive to minor perturbations, which is shown to be a more rigorous approach [19, 20, 22, 27]. In this way, it is similarly analogous to the coefficients of variation utilized in [17]. Works by [9, 22] have demonstrated that, in the context of linear regression-based model discovery, employing Bayesian evidence to enhance sparsity promotion significantly increases the likelihood of accurately selecting the optimal model in sparse and noisy datasets. This study will illustrate similar results in the context of the nonlinear optimization approach error-in-variable regression, by introducing the Orthogonal Distance Regression-based Bayesian Identification of Nonlinear Dynamics, a.k.a. ODR-BINDy, and comparing it with other STLS-based methodologies.

2.3 The ODR-BINDy algorithm, computational complexity and finding a good initial guess

Algorithm 1: ODR-BINDy algorithm: a greedy algorithm for Bayesian model selection based on nonlinear orthogonal distance regression to estimate parameters and uncertainties

input : Observation of all states $\hat{\mathbf{X}}$ over time, library functions $\boldsymbol{\Theta}(\mathbf{X})$, measurement noise variance $\boldsymbol{\sigma}_x^2$, measurement noise variance $\boldsymbol{\sigma}_{\partial t}^2$, prior parameter variance $\boldsymbol{\sigma}_p^2$
output : Learnt parameter Ξ with sparsity

initialization:
Linearly regress $L_I\boldsymbol{\Theta}(\hat{\mathbf{X}})$ against $L_{\partial t}\hat{\mathbf{X}}$ to get initial guess of Ξ. Perform multiple bootstrap regression to get multiple initial guesses;
Employ `lsqnonlin` to minimize (6) w.r.t. $(\mathbf{X}, \Xi)$. Use the above Ξ and $\hat{\mathbf{X}}$ as initial guesses. Save the most probable Ξ and denoised states $\mathbf{X}$;
repeat
  for *(parallelly) each non-zero* Ξ *and the corresponding column in* $\boldsymbol{\Theta}$ **do**
    Define $\tilde{\boldsymbol{\Theta}}$ that excludes the selected column during regression;
    Linearly regress $L_I\tilde{\boldsymbol{\Theta}}(\mathbf{X})$ against $L_{\partial t}\mathbf{X}$ with bootstrap to get a new initial guess of Ξ, using previous value of $\mathbf{X}$ before column removal.;
    Employ `lsqnonlin` to minimize (6) w.r.t. $(\mathbf{X}, \Xi)$, replacing $\boldsymbol{\Theta}(\mathbf{X})$ in (6) with the sparsified $\tilde{\boldsymbol{\Theta}}(\mathbf{X})$. Use the Ξ from boostrap regression and $\hat{\mathbf{X}}$ in previous step as initial guesses. Save the most probable Ξ and denoised states $\mathbf{X}$.;
    Use Gauss-Netwon and Laplace approximation, compute and save the Bayesian evidence;
  end
  Select the component in Ξ the removal of which yields the greatest evidence gain, and zero-out that component in Ξ and remove the corresponding column in $\boldsymbol{\Theta}$ for the later regressions against the corresponding target $L_{\partial t}\mathbf{X}$;
  Correspondingly, update $\mathbf{X}$ to the values obtained from the selected removal;
until *successive decline in evidence observed following the elimination of several terms*;
Select the model with the highest evidence, and the corresponding $(\mathbf{X}, \Xi)$, as the final output;

Following the Bayesian framework to select the model with maximum evidence, we designed the ODR-BINDy algorithm as outlined in algorithm 1. The reasoning behind the algorithm design is explained in more detail in the following paragraphs. First, we conduct a nonlinear optimization to minimize (6) by adjusting $(\mathbf{X}, \Xi)$ while incorporating all library terms. The data $\hat{\mathbf{X}}$ functions as the initial guess for $\mathbf{X}$, whereas multiple initial guesses for Ξ is provided by bootstrap linear regression. Our experience shows that the bootstrap strategy produce relatively good initial guesses, which is consistent with previous literature [8]. The randomness in bootstrap also gives multiple initial guesses, allowing the use of multi-start as a global optimization algorithm. Subsequently, in accordance with the greedy algorithm, we evaluate all models that possess one fewer term than the preceding iteration by executing nonlinear optimization to minimize (6) for each trial model and calculating the Bayesian evidence for each model. Upon evaluating all models with one fewer term, we select the model with the most evidence and proceed to the subsequent term elimination following the greedy technique. The algorithm terminates and selects the global maximum evidence model when it observes a successive decline in evidence following the elimination of several terms. The procedure consequently produces the trained sparse model along with its parameters and covariance, in addition to the associated (log-)Bayesian evidence that measures the model's quality.

Despite the apparent computational expense of executing many nonlinear optimizations, this phase, which constitutes the majority of the algorithm's duration, can be substantially accelerated by the following techniques:

(1) Supplying the analytical gradient to the optimizer can markedly accelerate computation. It is possible because the library function and its derivatives can be easily obtained. Furthermore, given that $L_{\partial t}$ and L_I often appear as sparse matrices, the gradient with respect to $\mathbf{X}$ is also sparse. Therefore, one can supply their sparsity pattern to the chosen quasi-Newton's method (`lsqnonlin` in this case) to substantially accelerate the optimization.
(2) In each successive trial of new models with one less term, the outcome $(\mathbf{X}, \Xi)$ from the preceding iteration with more terms is typically a reliable initial guess for the next nonlinear optimization. This allows successive optimization to be sped up significantly. Indeed, convergence using Newton's approach is often achieved within one hundred iterations if the initial guess is close enough.
(3) Conversely, if the trial model lacks the required term, the optimization frequently fails to converge. This prompts us to restrict the maximum number of steps during model trials and to retain the term whose elimination would result in non-convergence.
(4) The calculation of Bayesian evidence relies on the Laplace approximation, which only requires the parameter covariance at the minimum loss point. This computation can be further accelerated with the application of Gauss-Newton estimation, which also ensures the positive definiteness of the covariance. The full formulation of the evidence and Hessian is provided in § A.2, and the impact of Gauss-Newton estimation in place of full Hessian calculation can be found in § A.5.
(5) The computation of each trial model relies solely on the preceding run for an optimal initial guess and is independent of one another. Consequently, it can be further parallelized on a multi-node or multi-core CPU.

2.3.1 Choice of nonlinear optimization algorithm and computational complexity. As mentioned before, optimizing (6) is effectively a Ridge Orthogonal Distance Regression [4]. In this work we chose the `lsqnonlin` routine from `MATLAB` as the underlying optimization algorithm. This algorithm is a quasi-Newton's method with a trust region, specifically optimized for nonlinear least squares problems. It also allows specification of the analytical gradient, a feature we can fully exploit for acceleration due to the availability of such a gradient in our formulation. Since $L_{\partial t}$ and L_I are sparse banded matrices, the resulting Jacobian with respect to (X, Ξ) is also sparse (see left of fig. 3). As the computational cost of the quasi-Newton's method involves the formation and multiplication of this sparse Jacobian of size $\sim O(ND)$, the computational complexity of the nonlinear least-squares optimization scales with $O(ND)$. As a result, the overall computational complexity of the ODR-BINDy algorithm scales linearly with ND, as shown empirically in figure 3. To select a parsimonious model, the `lsqnonlin` routine is wrapped inside a greedy algorithm that removes terms sequentially based on the Bayesian evidence, as described above. The algorithm starts with the full library of M terms for each dimension, giving a total of MD terms to be tested. The greedy algorithm then performs trial least-squares optimization with each term eliminated, implying MD trial optimization in the first iteration. It then eliminates the term, the removal of which maximizes the gain in evidence, and reiterates the trial for the remaining $MD - 1$ terms. The elimination is then repeated, meaning the number of trial optimizations decreases by 1 for each iteration, leading to a total of $(MD)^2/2$ trial optimizations in the worst case. In practice, the number of trial optimizations is often slightly smaller than $(MD)^2/2$ as the algorithm terminates slightly earlier when it observes a successive decline in evidence following the elimination of several terms. Therefore, the overall computational complexity scales with $O((MD)^2 ND)$. However, since each trial optimization is independent of one another and only relies on the preceding run for an optimal

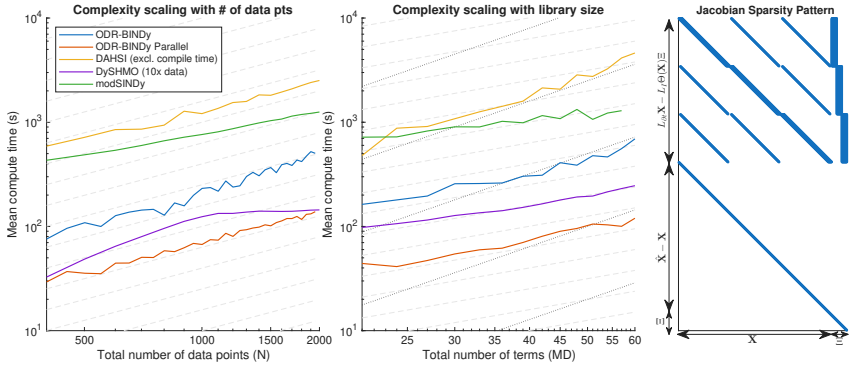


Fig. 3. Comparison of computational complexity of ODR-BINDy (w. or w/o 60-core parallelization), DAHSI, DySMHO, and modified-SINDy in identifying Lorenz63 from data with a 1% noise to signal ratio sampled at $\Delta t = 0.01$ (except for DySMHO, which requires 10 times more data by sampling at $\Delta t = 0.001$). Measurements of computational time are taken on an AMD EPYC 7742 CPU (left). Comparison of scaling in the number of data points from a second-order polynomial library. Note that DySMHO uses 10 times more data, but its scaling is plotted according to the equivalent time horizon used. (Middle) Comparison of scaling in library size. Note that modified-SINDy's library does not contain constant terms. In both plots, the dashed light grey lines represent linear scaling, whereas the dotted dark grey lines represent the quadratic scaling. (Right) Jacobian sparsity pattern of the loss function in (6) for a 3-dimensional system with 1000 data points and a second-order polynomial library.

initial guess, the algorithm can be further parallelized to achieve a $O(MD\ ND)$ scaling if the number of parallel processes used is larger than MD.

Figure 3 shows the empirical measurement of the overall computational complexity of the ODR-BINDy algorithm with and without parallelization, using data from the Lorenz63 system as an example (sampled at $\Delta t = 0.01$ for 1000 points and contaminated by a small measurement noise with 1% noise-to-signal ratio). As shown in figure 3, the computation time of ODR-BINDy shows a linear scaling with the number of data points N and a quadratic scaling with the library size M without parallelization as expected. (Note that at small MD, computational time of ODR-BINDy is more dominated by the initial guess rather than by the library size, so the scaling is not obvious until the library size becomes large.) However, as mentioned above, ODR-BINDy's greedy algorithm can be easily parallelized to achieve a linear scaling with M if the number of parallel processes used is larger than MD. Therefore, with enough parallel processes, ODR-BINDy can achieve a linear scaling with both N and M.

Figure 3 also compares ODR-BINDy against other aforementioned nonlinear algorithms for model discovery, including DAHSI, DySMHO and modified-SINDy. Although ODR-BINDy's linear scaling with N is slightly worse than modified-SINDy and DySMHO, which have sub-linear scaling with N due to modified-SINDy's arbitrary penalty of past loss and DySMHO's limited time window for assimilation, later results show that ODR-BINDy is more data efficient than modified-SINDy and DySMHO, which compensate for the slightly worse scaling in N. Meanwhile, given there exist enough parallel processes, the parallelized ODR-BINDy shows a linear scaling in M, which is comparable to modified-SINDy and DySMHO and much better than the quadratic scaling of DAHSI. Overall, figure 3 shows that with enough processors, parallelized ODR-BINDy is always faster than other existing methods.

It should be noted that the number of candidate functions M in the library, in theory, grows combinatorially with the number of dimensions D. In practice, the library is often limited by

truncation to a certain order or by prior knowledge, which can significantly reduce the number of candidate functions. However, it remains true that the computational cost scales at least quadratically with the number of dimensions D, which forbid the current approach and the other cited nonlinear approaches from being practical used in high-dimensional problems.

2.4 Hyperparameter tuning

In the present Bayesian framework, the hyperparameters governing the sparsity promotion process are the measurement variance σ_x^2, model error variance $\sigma_{\partial t}^2$, and prior σ_p^2. The inverses of these hyperparameters also serve as prefactors for their respective terms in the loss function. The measurement noise σ_x^2 is typically provided in real-world datasets (as a measurement error) or artificially introduced in synthetic datasets. To ensure proper model selection within the Bayesian framework, it is optimal to utilize its designated value. Meanwhile, the model error variance $\sigma_{\partial t}^2$ originates from the truncation error associated with the discretisation of the temporal domain or from mild stochasticity in the data. Discretisation error can be assessed based on the discretisation scheme and sample frequency; however, for the majority of real-world data, the minor stochasticity present is challenging to measure *a priori*. In this instance, it is advisable to conduct an additional hyperparameter optimization for $\sigma_{\partial t}^2$ to maximize the evidence [19].
Lastly, the prior variance σ_p^2, which represents prior knowledge regarding the magnitude of the learnt coefficients, should be sensibly large, i.e., larger in order of magnitude than the expected values of the coefficients such that the prior is weakly informing. As discussed in [9], the learning outcome is not sensitive to σ_p as long as it is in the appropriate order of magnitude, and its effect diminishes with an increasing number of data points. It is advisable to perform another hyperparameter optimization or hyperparameter sweep for σ_p^2 such that the evidence is maximized [19]; but, in practice, a reasonable value for σ_p^2 often produces outcomes very close to the optimized ones.

3 Comparison with other existing nonlinear model selection algorithms with examples

In the following section, we compare the performance of ODR-BINDy against the following existing algorithms that employ the nonlinear optimization approach (i.e., optimizing for both the parameter Ξ and the estimated denoised state $\mathbf{X}$), as well as Ensemble-SINDy [8], which exemplifies the linear regression-based approach. We will use synthetic data from known dynamical systems, specifically the Lorenz63, Rössler and Van Der Pol systems, as case studies to assess their efficacy in recovering information regarding the ground truth. Since the primary objective of this work is to reconstruct the governing equations from data, the principal metrics for assessment will be the success rates in recovering the correct sparsity pattern from a given polynomial dictionary. The respective model parameter errors are also calculated and shown in § A.1. While we acknowledge the existance of alternative measures (see [13, sec. 3.1]) for each algorithm's efficacy, such as the forecast prediction error and KL divergence from the time-averaged statistics, for the intent and purpose of this work, we will focus on comparing the success rates.

3.1 Brief summary of each algorithms

To establish a baseline for comparison, we briefly summarise the key features of each existing algorithm as follows:

- modified-SINDy [13]: The algorithm employs nonlinear optimization to integrate the data with an empirical penalty for historical loss, thereby alleviating the hard constraint problem, while concurrently learning the model and denoising the data. The algorithm alternates between nonlinear optimization (i.e., data assimilation) for parameter and initial state estimation, and

STLS regression to promote sparsity. The utilization of STLS results in the same drawback as SINDy, namely the empirical thresholding of coefficients that frequently discards significant terms with minimal coefficients. Furthermore, since the model is evaluated using ODE solvers, it may occasionally encounter finite-time blow up during optimization, leading to failure in convergence. The algorithm lacks robustness, and its performance is highly sensitive to the selection of hyperparameters and initial estimates.

- DySMHO [17]: A nonlinear optimization method utilizing a dynamic yet constrained time horizon strategy to address the issue of hard constraints. Sparsity is promoted not by thresholding based on the magnitude of coefficients, but by thresholding the variance of the coefficients across the ensemble of parameters acquired from each time window. The application of variance resembles the Bayesian principle in model selection; it seeks to eliminate terms with minimal posterior variance, which is associated with evidence. However, the moving horizon strategy restricts the algorithm to assimilating data within a finite time window, potentially constraining the information acquired from the data. If the data amount necessary for adequate model selection surpasses the time frame, the algorithm will be unable to discern the appropriate model. Furthermore, to obtain sufficient parameter sampling for an accurate estimation of parameter variance, the algorithm necessitates many sampling windows, which sets a minimum data requirement. Hence, DySMHO demands considerably more data than alternative algorithms to attain similar performance, as demonstrated in the subsequent examples.
- Ensemble-SINDy [8]: An linear regression-based approach utilizing bootstrapping to enhance noise robustness. Sparsity is promoted via STLS thresholding; however, the incorporation of bootstrapping renders the algorithm robust in its promotion of sparsity. Nonetheless, it remains a linear regression-based methodology, which constrains its efficacy in noisy datasets, as demonstrated in the subsequent examples.
- SIDDs [12]: A nonlinear optimization method employing a discretize-then-optimize strategy akin to ODR-BINDy, with the constraint (5) enforced as a KKT condition (hard constraint) during Newton's iteration. Sparsity is promoted by regularizers, and the quadratic programming problem is addressed using an iteratively reweighted least squares method. The method claims to achieve the Cramér-Rao lower bound in its parameter estimation process if the sparsity pattern is accurately identified; however, the rigid constraint may diminish the algorithm's robustness in initially identifying the correct sparsity pattern, as demonstrated in § A.4.
- DAHSI [25]: A nonlinear optimization approach with a variational annealing method in conjunction with a soft constrained loss similar to 6. Sparsity is promoted by STLS-styled thresholding followed by a AIC/BIC criterion to identify the Pareto front. Despite the correct use of the soft constraint, the STLS-AIC/BIC method for sparsity promotion is not as rigorous as the Bayesian evidence used in ODR-BINDy. Our experience shows that DAHSI is not robust in identifying the correct sparsity pattern, as shown in § A.3.

As illustrated above, although each method possesses distinct characteristics and occasionally exhibits similarities with ODR-BINDy—such as the application of soft constraints in DAHSI and the incorporation of Bayesian-like principles in DySMHO—none replicate the same amalgamation of features as ODR-BINDy. ODR-BINDy is also unique in its treatment of noise and model selection from the Bayesian perspective. The subsequent examples demonstrate that this appropriate combination of features enables ODR-BINDy to surpass other existing methods in accurately recovering the correct sparsity pattern in noisy and sparse datasets.

3.2 Adjustments and fine tuning of each algorithms in the test examples

To our best effort, we have tried to tune the hyperparameters of each algorithm according to each algorithm's documentation to maximize their performance. When an example is present in the algorithm's repo, we endeavor to adhere to the default configurations from the algorithm, presuming they are optimized for peak performance. Moreover, to optimize the performance of each existing algorithm, we have chosen not to evaluate them with entirely uniform test sets. Instead, we have made small adjustments in the sampling rate and candidate library to ensure optimal performance from each algorithm. The following are the specific adjustments made for DySMHO, Ensemble-SINDy and modified-SINDy:

- DySMHO: We see that DySMHO necessitates significantly more data to achieve comparable noise robustness to the other approaches; hence, we have provided DySMHO with data sampled at a frequency **ten times** that of other methods. Additionally, DySMHO depends on the `pyomo` algorithm, which interfaces with optimization libraries like AMPL. Although the DySMHO utilized GAMS' version of `CONOPT`, we discovered that substituting `CONOPT` with `IPOPT` had minimal overall performance effect (see § A.6). Owing to the constraints of the proprietary license regarding the permissible number of executions, we have selected `IPOPT` for the subsequent comparison.
- Ensemble-SINDy and modified-SINDy: The existing implementation uses a library that does not contain the constant term for the Lorenz63 example. We have tested the inclusion of the constant term within the library; however, this action substantially diminishes the performance of the algorithms. Consequently, we adhere to the configuration provided by the algorithms and utilize solely the library without the constant term for Lorenz63.

We do note that, by using a slightly different dictionary for Ensemble/modified-SINDy and a higher sampling frequency for DySMHO, these algorithms confer certain advantages over others. Hence, the following comparison should not be read as a fair benchmark **between** existing algorithms. Instead, the comparison shows that, despite favouring metrics, these existing algorithms do not outperform ODR-BINDy in recovering the correct sparsity, i.e., the correct model, in noisy and sparse datasets.
We also attempted to test the performance of DAHSI and SIDDs; however, we found it difficult to make relevant performance comparison with these two algorithms due to the following reasons:

- SIDDs: Despite our utmost efforts, we are unable to execute the algorithm from their open-source repository due to absent dependencies; nonetheless, it is noteworthy that they cited better performance over modified-SINDy [12]. The appendix § A.4 illustrates the comparison of SIDDs with ODR-BINDy based on our best approximation to the performance they've cited.
- DAHSI: DAHSI exhibits insufficient performance to be relevant under the noise regime examined in this study; hence, a comprehensive comparison with DAHSI was not conducted. Sec. A.3 illustrates the comparison between DAHSI and ODR-BINDy.

3.3 Van Der Pol

Fig. 4 shows Van Der Pol system

$$\dot{x} = y, \qquad \dot{y} = \mu(1 - x^2)y - x, \qquad [x(0), y(0)] = [-2, 1]$$

where $\mu = 0.5$. Sampled at $\Delta t = 0.01$ ($\Delta t = 0.001$ for DySMHO) till the time T that specifies the data length, the data is then contaminated with an i.i.d. Gaussian Noise $\mathcal{N}(0, \sigma_x)$ before being fed into each algorithm to recover the governing equation and parameters based on a 3rd order polynomial dictionary.

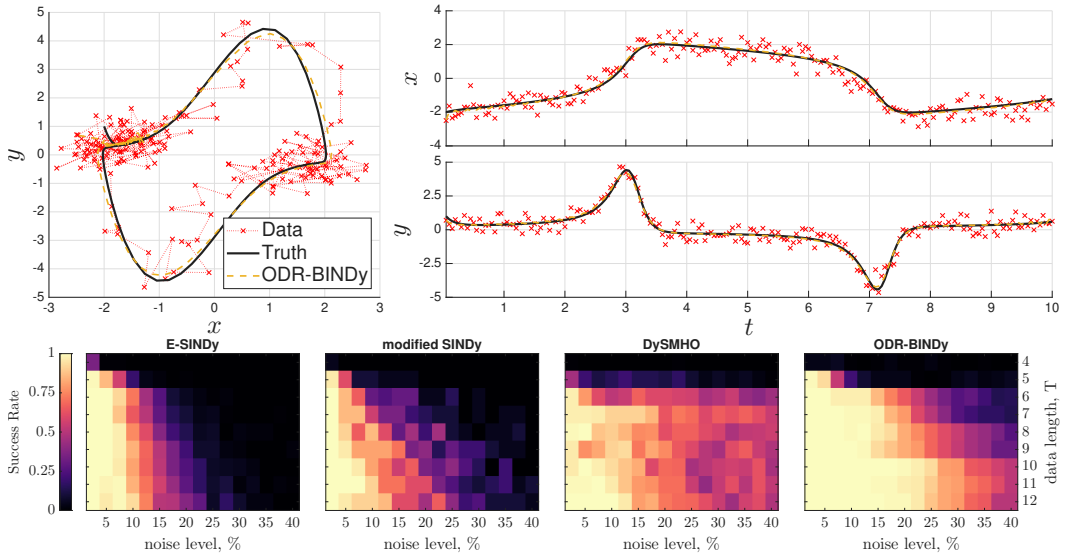


Fig. 4. (above) Example of recovering the Van Der Pol system and denoising the data by ODR-BINDy. The data here contain 30% noise and sampled at $\Delta t = 0.01$. (below) Success Rate of E-SINDy, modified-SINDy, DySMHO and ODR-BINDy in recovering the Van Der Pol system from data sampled at $\Delta t = 0.01$ (except for DySMHO, which has $\Delta t = 0.001$).

Fig. 4 shows the performance of each algorithm in recovering the Van Der Pol system across data length, given by the time T, and the noise level. The noise level in the plot is defined as the ratio σ_x/σ_{Signal}, where σ_{Signal} is the standard deviation of the flatten noiseless dataset. We see that linear-based Ensemble-SINDy is not as noise-robust as other nonlinear optimization-based approaches. This is largely due to the nonlinear approaches' ability to denoise data. Among the nonlinear methods, ODR-BINDy is more noise robust and data efficient than other methods. Meanwhile, DySMHO may seem slightly better performing than modified-SINDy in a high noise regime, but note that DySMHO is using ten times more data points to achieve comparable performance with modified-SINDy and ODR-BINDy. This is likely due to the fact that it uses the weak form collocation method to discretise the time domain. The weak form tends to work well if the time window is highly sampled but may also filter the critical signal in the high frequency if the sampling frequency is not significantly higher than the signal [9].

3.4 Lorenz 63

Fig. 5 shows the typical Lorenz63 system

$$\dot{x} = \sigma(x - y), \qquad \dot{y} = x(\rho - z) - y, \qquad \dot{z} = xy - \beta z \qquad [x(0), y(0), z(0)] = [-8, 8, 27]$$

with $\sigma = 10, \rho = 28, \beta = 8/3$, and the performance of each algorithm in recovering the system from data sampled as $\Delta t = 0.01$ ($\Delta t = 0.001$ for DySMHO), contaminated by Gaussian noise. A 2nd order polynomial library is used, but as described above, an exception was made for Ensemble-SINDy and modified-SINDy to remove the constant term from the library to improve their convergence. Again, we see that linear-based Ensemble-SINDy is not as noise-robust as the nonlinear optimization-based approaches. However, modified-SINDy and DySMHO are also not performing as well as in the Van Der Pol system. This can be attributed to the fact that Lorenz63 is chaotic, and therefore modified-SINDy and DySMHO cannot fully make use of the whole dataset in recovering the equation. Instead,

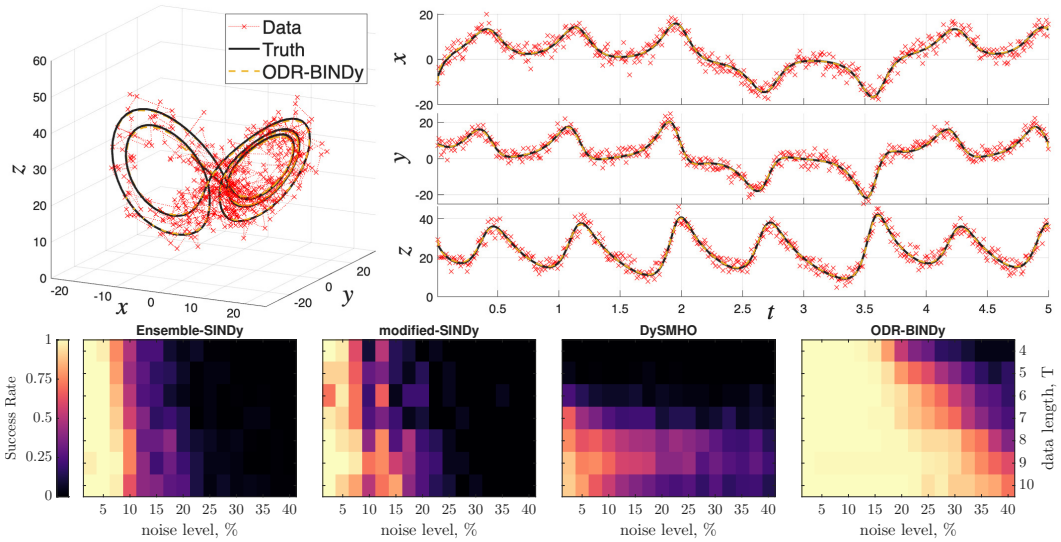


Fig. 5. (above) Example of recovering the Lorenz63 system and denoising the data by ODR-BINDy. The data here contain 20% noise and sampled at $\Delta t = 0.01$. The Lyapunov time of the system is approximated to be 1.07. (below) Success Rate of E-SINDy, modified-SINDy, DySMHO and ODR-BINDy in recovering the Lorenz63 system from data sampled at $\Delta t = 0.01$ (except for DySMHO, which has $\Delta t = 0.001$).

DySMHO limits the scope of the data by learning from a limited horizon, while modified-SINDy penalises losses from data too far behind or into the future from the current time. Both of these strategies limit the information one can gain from the data and therefore limit noise robustness and data efficiency. In contrast, ODR-BINDy can assimilate the entire time series to learn the model and can therefore maximize the information one can extract from the data regarding the parameter values and model selection.

3.5 Rössler

Fig. 6 shows the same comparison in the Rössler system

$$\dot{x} = -y - z, \qquad \dot{y} = x + ay, \qquad \dot{z} = b + z(x - c), \qquad [x(0), y(0), z(0)] = [-6, 5, 0]$$

where $a = 0.2, b = 0.2, c = 5.7$. The data is sampled at $\Delta t = 0.05$, and contaminated by Gaussian noise scaled by the noise level. In this case, all algorithm are attempting to learn the model from a 2nd order polynomial library with constant term. Each algorithm's performance is shown in lower half of fig. 6. Similar to the results before, ODR-BINDy is more noise-robust and data-efficient than other methods.

4 Limitations

Although ODR-BINDy generally shows better performance than other existing algorithms in recovering the correct model, especially in data contaminated by large measurement noise, there remain some limitations to ODR-BINDy:

- The performance of ODR-BINDy is not as good in decaying signals, such as damped cubic oscillators. This is due to the fact that the discretisation error is currently modelled as a time-invariant stochastic noise, while a decaying signal will lead to a decaying discretisation error that temporally correlates with the signal itself. Such time-varying and signal-correlating errors

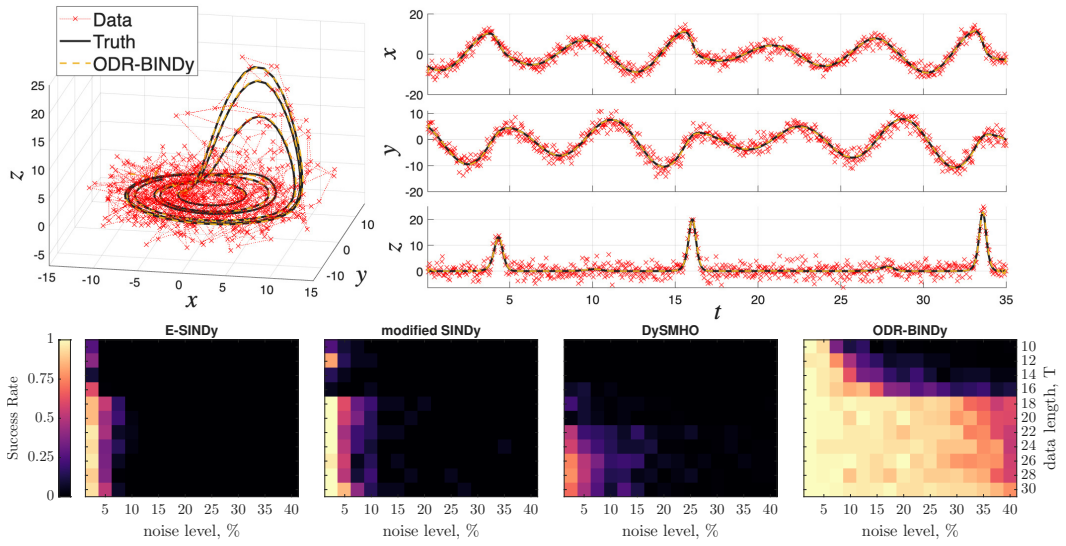


Fig. 6. (above) Example of recovering the Rössler system and denoising the data by ODR-BINDy. The data here contain 40% noise and sampled at $\Delta t = 0.05$. The Lyapunov time in the system is approximated to be 14.16. (below) Success Rate of E-SINDy, modified-SINDy, DySMHO and ODR-BINDy in recovering the Rössler system from data sampled at $\Delta t = 0.05$ (except for DySMHO, which has $\Delta t = 0.005$).

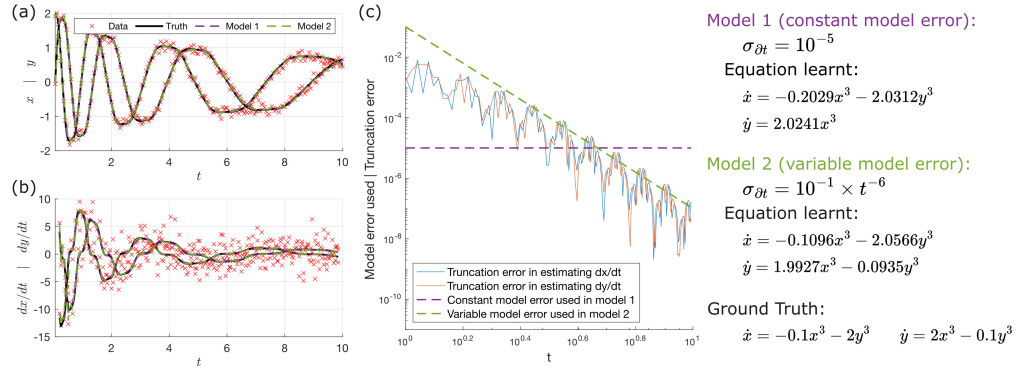


Fig. 7. Attempts to learn the equation of the nonlinear oscillator using ODR-BINDy. Here, $\Delta t = 0.05$ and $\sigma_x = 0.084$ (10% noise). Since both (a) the states and (b) the time derivatives are decaying, (c) the truncation error is also decaying in time. Here, we compare two models, with different hyperparameter settings for $\sigma_{\partial t}$ to demonstrate the potential issue in learning from decaying signals. Model 1 (purple) represents the typical hyperparameter setting for $\sigma_{\partial t}$, which is not consistent with the real truncation error. Model 2 (green) assumes that the trend of the truncation error is somehow known *a priori*. Although both models seemingly able to follow the signal in $(a - b)$, the equation learnt by model 1 is oversimplified, likely owing to the fact that the noise is overestimated later in time. Only by modifying the noise to follow the real truncation error in model 2 can the correct model be inferred, but in practice the truncation error is likely not known *a priori*.

could lead to bias in the model and thereby prevent the Bayesian algorithm from selecting the correct model. Fig. 7 shows an example of ODR-BINDy failing to recover the cubic oscillator equation from its decaying signal.

- Currently, ODR-BINDy is not designed to directly provide forecasts of the time-series data. Works are ongoing in demonstrating ODR-BINDy's capability for direct forecasting, but there are some technical challenges in ensuring the statistical guarantee of the forecast as one switches from the finite difference time discretisation to conventional ODE solvers for the forecasting. There is also the subtle challenge of how one should determine the distribution of the initial condition for the forecast from the given historical data. With that said, one can always apply classical data assimilation techniques [1] on the model/sparsity learnt from ODR-BINDy to perform forecasting. The procedure of this two-step strategy will be similar to many classical works on data assimilation and forecasting, so we shall not repeat them here.

5 Conclusion and future work

This study demonstrates the efficacy of ODR-BINDy in identifying governing equations of a dynamical system from noisy temporal state observations. It has not only exhibited superior noise robustness and data efficiency compared to all current SINDy variants in accurately recovering the correct model, but its speed also surpasses that of all existing SINDy variants that utilize nonlinear optimization to address the error-in-variable problem. A portion of the performance improvement can presumably be ascribed to the Bayesian framework, which offers a robust basis for model selection and the enhancement of sparsity. However, the chief breakthrough that gives rise to the algorithm's speed and capacity to assimilate long time series even in chaotic systems is due to stochastic relaxation. Stochastic relaxation not only mitigated the sensitivity issue associated with conventional parameter estimation in chaotic systems but also underscored the significance of truncation errors in data assimilation. Furthermore, it elucidated the origins of the scaling factors for each component of the total loss function, providing both intuitive guidance for tuning and theoretical rationale for the associated hyperparameters.

Despite the algorithm's demonstrated efficiency, partly due to the well-optimized `lsqnonlin` algorithm and the implementation of sparse and analytical gradients, there exists considerable potential for enhancement, particularly regarding its computing speed. The existing greedy approach exhibits a complexity of $O(NDMD)$ with parallelization, where M represents the number of candidate functions in the library and increases exponentially with D. The computational complexity in MD can be further reduced if we incorporate analytical estimation of the potential change in evidence for the removal testing process, akin to the strategy used by the `SparseBayes` method [27]. This improvement will be essential for high-dimensional applications, such as the recovery of dynamics in reduced-order models for fluid problems [15]. Alternatively, employing gradient descent methods such as Adam in place of the existing multi-start trust-region approach may enhance performance if the initial guess does not give satisfactory results.

Besides potential improvement in computational speed, there are many aspects where the method can be further improved. For instance, although we have solely illustrated the method's efficacy utilizing an equispaced finite difference discretisation, numerous collocation methods, including the weak formulation, exist to discretise the equation (both equi-spatially and non-equispatially) that can also be represented as the matrices $L_{\partial t}$ and L_I. Consequently, one can readily interchange these matrices with those associated with the preferred collocation or discretisation technique. Alternatively, one may pursue high-order ODE solvers to impose the equation constraint, as exemplified by modified-SINDy [13], rather than use the discretize-then-optimize methodology. Nevertheless, additional effort is necessary to integrate stochastic relaxation into this methodology.

Moreover, our formulation in (6) can be readily adapted to accommodate non-Gaussian noise. In particular, further research is required to examine the actual noise distribution resulting from truncation errors caused by discretisation. Although relaxing the noise model to a non-Gaussian distribution will lead to a non-Gaussian posterior, in the usual case where the number of data points far exceeds the number of parameters, the posterior is most likely unimodal and sub-Gaussian. Hence, the Gauss-Newton-based Laplace approximation of the evidence is likely a good approximation for most cases.
Lastly, the nonlinear optimization of our current algorithm also allows for a more extensible dictionary. Indeed, part of DySMHO [17] and DAHSI [25] have already demonstrated learning a parameterised library, and extending the current algorithm to include an implicit term is also possible. DAHSI [25] have also demonstrated the possibility of recovering the full equations with some of the state variable measurements hidden from the algorithm. It would be interesting to investigate if ODR-BINDy is capable of similar problems, although it is unclear if the algorithm can perform just as well without the observational data as part of the initial condition.

Acknowledgments

We would like to to thank U. Fasel, A. Novoa, L. Magri, B. Moseley and M. P. Juniper for their feedback on the manuscript, and J. N. Kutz for the brief but fruitful discussion.
L. F. acknowledges the support from the Imperial College Research Fellowship. This project is supported by Schmidt Sciences, LLC.

Code availability

The code for ODR-BINDy can be found on github.com/llfung/ODR-BINDy. The code to generate the comparisons with other methods can be found on the following links: E-SINDy, modified-SINDy and DySMHO.

Data availability

The data on the success rate and model error coefficients can be found on Zenodo.

References

[1] Marc Asch, Marc Bocquet, and Maëlle Nodet. 2016. *Data Assimilation: Methods, Algorithms, and Applications.* Number 11 in Fundamentals of Algorithms. SIAM, Philadelphia, Pennsylvania. doi:10.1137/1.9781611974546
[2] Yonathan Bard. 1974. *Nonlinear Parameter Estimation.* Academic Press, New York.
[3] Mohamed Aziz Bhouri and Paris Perdikaris. 2022. Gaussian processes meet NeuralODEs: a Bayesian framework for learning the dynamics of partially observed systems from scarce and noisy data. *Phil. Trans. R. Soc. A* 380, 2229 (June 2022), 20210201. doi:10.1098/rsta.2021.0201
[4] Paul T. Boggs, Richard H. Byrd, and Robert B. Schnabel. 1987. A Stable and Efficient Algorithm for Nonlinear Orthogonal Distance Regression. *SIAM J. Sci. Stat. Comput.* 8, 6 (Nov. 1987), 1052–1078. doi:10.1137/0908085
[5] David M. Bortz, Daniel A. Messenger, and Vanja Dukic. 2023. Direct Estimation of Parameters in ODE Models Using WENDy: Weak-Form Estimation of Nonlinear Dynamics. *Bull. Math. Biol.* 85, 11 (Oct. 2023), 110. doi:10.1007/s11538-023-01208-6
[6] Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. 2016. Discovering Governing Equations from Data by Sparse Identification of Nonlinear Dynamical Systems. *Proc. Natl. Acad. Sci. U.S.A.* 113, 15 (April 2016), 3932–3937. doi:10.1073/pnas.1517384113
[7] Kathleen Champion, Bethany Lusch, J. Nathan Kutz, and Steven L. Brunton. 2019. Data-Driven Discovery of Coordinates and Governing Equations. *Proc. Natl. Acad. Sci. U.S.A.* 116, 45 (Nov. 2019), 22445–22451. doi:10.1073/pnas.1906995116
[8] Urban Fasel, J. Nathan Kutz, Bingni W. Brunton, Brunton, and Steven L. Brunton. 2022. Ensemble-SINDy: Robust Sparse Model Discovery in the Low-Data, High-Noise Limit, with Active Learning and Control. *Proc. R. Soc. A* 478, 2260 (April 2022), 20210904. doi:10.1098/rspa.2021.0904
[9] Lloyd Fung, Urban Fasel, and Matthew Juniper. 2025. Rapid Bayesian Identification of Sparse Nonlinear Dynamics from Scarce and Noisy Data. *Proc. R. Soc. A.* 481, 2307 (Feb. 2025), 20240200. doi:10.1098/rspa.2024.0200

[10] L. Mars Gao, Urban Fasel, Steven L. Brunton, and J. Nathan Kutz. 2023. *Convergence of Uncertainty Estimates in Ensemble and Bayesian Sparse Model Discovery.* arXiv:2301.12649 [cs, math, stat]
[11] Seth M. Hirsh, David A. Barajas-Solano, and J. Nathan Kutz. 2022. Sparsifying Priors for Bayesian Uncertainty Quantification in Model Discovery. *Royal Soc. Open Sci.* 9, 2 (Feb. 2022), 211823. doi:10.1098/rsos.211823
[12] Jeffrey M. Hokanson, Gianluca Iaccarino, and Alireza Doostan. 2023. Simultaneous Identification and Denoising of Dynamical Systems. *SIAM J. Sci. Comput.* 45, 4 (July 2023), A1413–A1437. doi:10.1137/22M1486303
[13] Kadierdan Kaheman, Steven L Brunton, and J Nathan Kutz. 2022. Automatic Differentiation to Simultaneously Identify Nonlinear Dynamics and Extract Noise Probability Distributions from Data. *Mach. Learn.: Sci. Technol.* 3, 1 (March 2022), 015031. doi:10.1088/2632-2153/ac567a
[14] Kadierdan Kaheman, J. Nathan Kutz, and Steven L. Brunton. 2020. SINDy-PI: A Robust Algorithm for Parallel Implicit Sparse Identification of Nonlinear Dynamics. *Proc. R. Soc. A* 476, 2242 (Oct. 2020), 20200279. doi:10.1098/rspa.2020.0279
[15] Alan A. Kaptanoglu, Jared L. Callaham, Aleksandr Aravkin, Christopher J. Hansen, and Steven L. Brunton. 2021. Promoting Global Stability in Data-Driven Models of Quadratic Nonlinear Dynamics. *Phys. Rev. Fluids* 6, 9 (Sept. 2021), 094401. doi:10.1103/PhysRevFluids.6.094401
[16] George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. 2021. Physics-Informed Machine Learning. *Nat. Rev. Phys* 3, 6 (June 2021), 422–440. Issue 6. doi:10.1038/s42254-021-00314-5
[17] Fernando Lejarza and Michael Baldea. 2022. Data-Driven Discovery of the Governing Equations of Dynamical Systems via Moving Horizon Optimization. *Sci. Rep.* 12, 1 (July 2022), 11836. doi:10.1038/s41598-022-13644-w
[18] Lennart Ljung. 2010. Perspectives on System Identification. *Annu. Rev. Control* 34, 1 (April 2010), 1–12. doi:10.1016/j.arcontrol.2009.12.001
[19] David J. C. MacKay. 1992. Bayesian Interpolation. *Neural Comput.* 4, 3 (May 1992), 415–447. doi:10.1162/neco.1992.4.3.415
[20] David J. C. MacKay. 2003. *Information Theory, Inference, and Learning Algorithms.* Cambridge University Press, Cambridge, UK ; New York.
[21] Daniel A. Messenger and David M. Bortz. 2021. Weak SINDy for Partial Differential Equations. *J. Comput. Phys.* 443 (2021), 110525. doi:10.1016/j.jcp.2021.110525
[22] Robert K. Niven, Laurent Cordier, Ali Mohammad-Djafari, Markus Abel, and Markus Quade. 2024. Dynamical System Identification, Model Selection, and Model Uncertainty Quantification by Bayesian Inference. *Chaos* 34, 8 (Aug. 2024), 083140. doi:10.1063/5.0200684
[23] James O. Ramsay, Giles Hooker, David Campbell, and Jiguo Cao. 2007. Parameter Estimation for Differential Equations: A Generalized Smoothing Approach. *J. R. Stat. Soc. B* 69, 5 (Nov. 2007), 741–796. doi:10.1111/j.1467-9868.2007.00610.x
[24] Patrick A. K. Reinbold, Daniel R. Gurevich, and Roman O. Grigoriev. 2020. Using Noisy or Incomplete Data to Discover Models of Spatiotemporal Dynamics. *Phys. Rev. E* 101, 1 (jan 2020), 010203. doi:10.1103/PhysRevE.101.010203
[25] Helena Ribera, Sasha Shirman, Anh V. Nguyen, and Niall M. Mangan. 2022. Model Selection of Chaotic Systems from Data with Hidden Variables Using Sparse Data Assimilation. *Chaos* 32, 6 (June 2022), 063101. doi:10.1063/5.0066066
[26] Samuel H. Rudy, Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. 2017. Data-Driven Discovery of Partial Differential Equations. *Sci. Adv.* 3, 4 (April 2017), e1602614. doi:10.1126/sciadv.1602614
[27] Michael E. Tipping and Anita C. Faul. 2003. Fast Marginal Likelihood Maximisation for Sparse Bayesian Models. In *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. R4)*, Christopher M. Bishop and Brendan J. Frey (Eds.). PMLR, 276–283.
[28] Sheng Zhang and Guang Lin. 2018. Robust Data-Driven Discovery of Governing Physical Laws with Error Bars. *Proc. R. Soc. A* 474, 2217 (Sept. 2018), 20180305. doi:10.1098/rspa.2018.0305
[29] Peng Zheng, Travis Askham, Steven L. Brunton, J. Nathan Kutz, and Aleksandr Y. Aravkin. 2019. A Unified Framework for Sparse Relaxed Regularized Regression: SR3. *IEEE Access* 7 (Jan. 2019), 1404–1423. doi:10.1109/ACCESS.2018.2886528

A Appendix

A.1 Averaged model parameter errors

Fig. 8, fig. 9 and fig. 10 show the averaged model parameter error from the learning result of Ensemble-SINDy, modified-SINDy, DAHSI and ODR-BINDy. The model parameter error of each run is defined by the normalised difference between the parameter value learnt against the ground truth parameter value Ξ_{true}, i.e.

$$\frac{||\Xi - \Xi_{true}||_F}{||\Xi_{true}||_F}$$

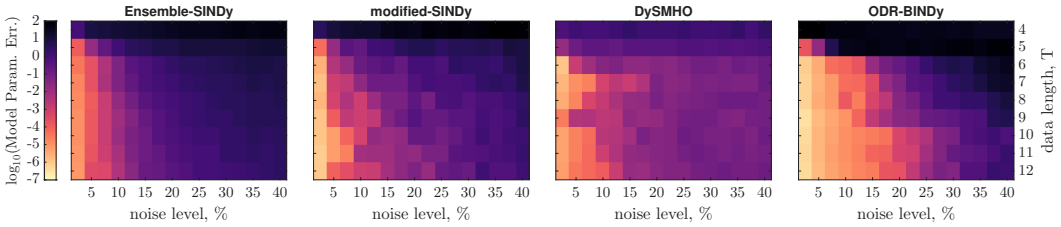


Fig. 8. Van Der Pol: Comparison of model parameter error from the parameter values learnt by E-SINDy, modified-SINDy, DySMHO and ODR-BINDy at increasing noise-to-signal ratio and decreasing data length. Hyperparmeter and settings are the same as that of fig. 4 for each algorithm.

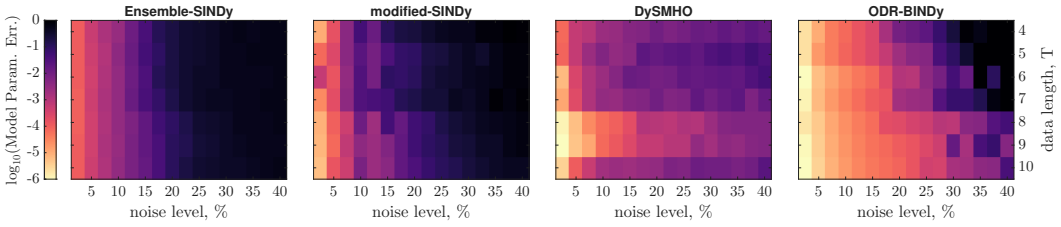


Fig. 9. Lorenz63: Comparison of model parameter error from the parameter values learnt by E-SINDy, modified-SINDy, DySMHO and ODR-BINDy at increasing noise-to-signal ratio and decreasing data length. Hyperparmeter and settings are the same as that of fig. 5 for each algorithm.

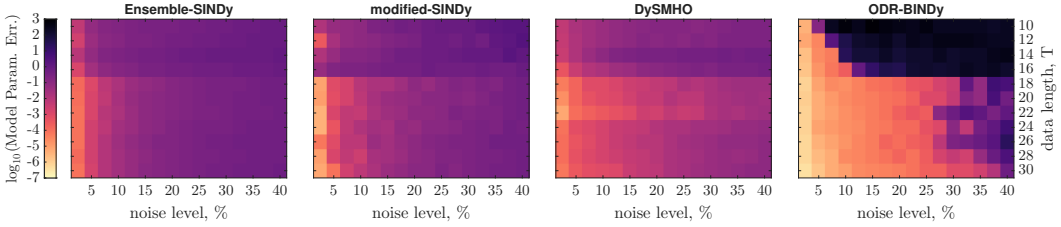


Fig. 10. Rössler: Comparison of model parameter error from the parameter values learnt by E-SINDy, modified-SINDy, DySMHO and ODR-BINDy at increasing noise-to-signal ratio and decreasing data length. Hyperparmeter and settings are the same as that of fig. 6 for each algorithm.

and the plots show the averaged value over many runs. (Note that despite the Frobenius norm used here being different from the 2-norm used in (17) of [13], the vector Ξ in [13] was assumed to be flatten. Hence, the parameter error defined here is the same as (17) in [13].) The hyperparameter setting are the same as the one used in previous sections, and each plot corrsponds to the respective success rate plot in fig. 4, fig. 5 and fig. 6.

A.2 Formula for the Laplace Approximation in ODR

In Orthogonal Distance Regression (ODR), we seek to minimize the loss function as specified in (6) by varying $\mathbf{X}$ and Ξ. To facilitate easier notation in the following derivation, we define $\eta_{ij} = \sum_{k,n}[L_{\partial t}]_{ik}X_{kj} - [L_I]_{ik}\Theta_{kn}\Xi_{nj}$ and $\zeta_{ij} = X_{ij} - \hat{X}_{ij}$. Therefore, (6) can be rewritten in index

form as

$$\mathcal{L}(\Xi, \mathbf{X}) = \frac{1}{2}\left[\sum_{id}\frac{\zeta_{id}^2}{\sigma_x^2} + \sum_{id}\frac{\eta_{id}^2}{\sigma_{\partial t}^2} + \sum_{mn}\frac{\Xi_{mn}^2}{\sigma_p^2}\right]$$

In particular, by minimizing the above against $\mathbf{X}$, we are also effectively minimizing $\zeta_{ij}^2/\sigma_x^2 + \eta_{ij}^2/\sigma_{\partial t}^2$, the "rescaled" distance between the data points $\hat{\mathbf{X}}$ and the respective point $\mathbf{X}$ on the regressed curve, the rescaling of which is defined by σ_x and $\sigma_{\partial t}$ on the data error space (spanned by $\boldsymbol{\zeta}$) and model error space (spanned by $\boldsymbol{\eta}$) respectively. The resulting optimal points $\mathbf{X} = \mathbf{X}^*(\Xi)$ is the "rescaled closest points" on the curve to the respective data point $\hat{\mathbf{X}}$ that not only minimizes $\mathcal{L}(\Xi, \mathbf{X})$ for a given curve specified by Ξ, they can also be interpreted as the corresponding estimated denoised states.

To help with the notation and the following derivation, we shall define the following derivatives:

$$\eta_{id} = \sum_{k,n}[L_{\partial t}]_{ik}X_{kd} - [L_I]_{ik}\Theta_{kn}\Xi_{nd}$$

$$\frac{\partial \eta_{id}}{\partial X_{je}} = \sum_{k,n}[L_{\partial t}]_{ij}\delta_{de} - [L_I]_{ik}\frac{\partial \Theta_{kn}}{\partial X_{je}}\Xi_{nd}$$

$$\frac{\partial^2 \eta_{id}}{\partial X_{je}\partial X_{lf}} = \sum_{k,n} -[L_I]_{ik}\frac{\partial \Theta_{kn}}{\partial X_{je}\partial X_{lf}}\Xi_{nd}$$

$$\frac{\partial \eta_{id}}{\partial \Xi_{ne}} = \sum_{k} -[L_I]_{ik}\Theta_{kn}\delta_{de} = \frac{\partial \eta_{id}}{\partial \Xi_{nd}}$$

$$\frac{\partial^2 \eta_{id}}{\partial \Xi_{ne}\partial X_{jf}} = \sum_{k} -[L_I]_{ik}\frac{\partial \Theta_{kn}}{\partial X_{jf}}\delta_{de} = \frac{\partial^2 \eta_{id}}{\partial \Xi_{nd}\partial X_{jf}}.$$

Now, since $\mathbf{X}^*$ is the optimal point, one can show that at $\mathbf{X} = \mathbf{X}^*(\Xi)$ and any Ξ,

$$\left.\frac{\partial \mathcal{L}}{\partial X_{je}}\right|_{\mathbf{X}=\mathbf{X}^*,\Xi} = \frac{\zeta_{je}}{\sigma_x^2} + \frac{\eta_{id}}{\sigma_{\partial t}^2}\frac{\partial \eta_{id}}{\partial X_{je}} = \mathbf{0} \tag{7}$$

This is the orthogonal constrain that underly ODR [4, see fig. 1-2], which has a geometrical interpretation as follow: at the optimal point $\mathbf{X} = \mathbf{X}^*$ that minimizes the rescaled distance, the rescaled distance vector $[\boldsymbol{\zeta}/\sigma_x; \boldsymbol{\eta}/\sigma_{\partial t}]$ must be orthogonal to the rescaled local slope of the curve at $\mathbf{X}^*$, which is $[\mathbf{1}/\sigma_x; (\partial\boldsymbol{\eta}/\partial\mathbf{X})/\sigma_{\partial t}]$. While in modern implementation of ODR, we no longer use the above constrain to compute $\mathbf{X}^*$, here, we shall assume that when the nonlinear regression converges to the optimal point, the solution must also satisfy (7). We shall use this fact as the basis to derive the analytical expression of the Hessian needed to compute the Bayesian evidence.

Since the optimization to minimize $\mathcal{L}$ will inevitably gives a solution where $\mathbf{X} = \mathbf{X}^*$, from now on, we will assume the optimal $\mathbf{X}$ is computed. Here, we further simplify the expression of all terms that is computed at $\mathbf{X} = \mathbf{X}^*(\Xi)$ with the * notation, e.g. $\eta_{id}^* = \eta_{id}|_{X=X^*}$.

Differentiating (7) by Ξ gives

$$\begin{aligned}&\left(\frac{1}{\sigma_x^2}\delta_{de}\delta_{ij} + \frac{1}{\sigma_{\partial t}^2}\left(\frac{\partial \eta_{kf}^*}{\partial X_{je}}\frac{\partial \eta_{kf}^*}{\partial X_{id}} + \eta_{kf}^*\frac{\partial^2 \eta_{kf}^*}{\partial X_{je}\partial X_{id}}\right)\right)\frac{dX_{je}^*}{d\Xi_{ne}}\\ =&\frac{-1}{\sigma_{\partial t}^2}\left(\frac{\partial \eta_{kf}^*}{\partial \Xi_{ne}}\frac{\partial \eta_{kf}^*}{\partial X_{id}} + \eta_{kf}^*\frac{\partial^2 \eta_{kf}^*}{\partial \Xi_{ne}\partial X_{id}}\right),\end{aligned} \tag{8}$$

which we can solve for $dX^*_{je}/d\Xi_{ne}$ numerically upon finding $\mathbf{X}^*$ from the nonlinear optimization. With $dX^*_{je}/d\Xi_{ne}$ and the above derivatives computed, we are now ready to calculate the Bayesian evidence.

A.2.1 Computing the evidence by Gauss-Newton Laplace approximation. Now, to compute the Bayesian evidence by Laplace approximation, we require the Hessian

$$\frac{d^2\mathcal{L}}{d\Xi_{nd}d\Xi_{me}} = \frac{1}{\sigma^2_{\partial t}}\left[\eta^*_{kf}\left(\frac{\partial^2\eta^*_{kf}}{\partial\Xi_{nd}\partial X_{ig}}\frac{dX^*_{ig}}{d\Xi_{me}} + \frac{\partial^2\eta^*_{kf}}{\partial\Xi_{me}\partial X_{ig}}\frac{dX^*_{ig}}{d\Xi_{nd}} + \frac{\partial\eta^*_{kf}}{\partial X_{ig}}\frac{d^2X^*_{ig}}{d\Xi_{nd}d\Xi_{me}}\right)\right. \tag{9}$$

$$\left. + \left(\frac{\partial\eta^*_{kf}}{\partial X_{ig}}\frac{dX^*_{ig}}{d\Xi_{nd}} + \frac{\partial\eta^*_{kf}}{\partial\Xi_{nd}}\right)\left(\frac{\partial\eta^*_{kf}}{\partial X_{ig}}\frac{dX^*_{ig}}{d\Xi_{me}} + \frac{\partial\eta^*_{kf}}{\partial\Xi_{me}}\right)\right] \tag{10}$$

$$+ \frac{1}{\sigma^2_x}\left[(X_{ig} - \hat{X}_{ig})\frac{d^2X^*_{ig}}{d\Xi_{nd}d\Xi_{me}} + \frac{dX^*_{ig}}{d\Xi_{nd}}\frac{dX^*_{ig}}{d\Xi_{me}}\right] + \frac{1}{\sigma^2_p}\delta_{nm}\delta_{de} \tag{11}$$

which involves expensive computation of $d^2\mathbf{X}/d\Xi d\Xi$. The Gauss-Newton approximation allows us to approximate the above Hessian without computing $d^2\mathbf{X}/d\Xi d\Xi$, by effectively ignoring the second order derivatives, giving

$$\frac{d^2\mathcal{L}}{d\Xi_{nd}d\Xi_{me}} \approx \frac{1}{\sigma^2_{\partial t}}\left[\left(\frac{\partial\eta^*_{kf}}{\partial X_{ig}}\frac{dX^*_{ig}}{d\Xi_{nd}} + \frac{\partial\eta^*_{kf}}{\partial\Xi_{nd}}\right)\left(\frac{\partial\eta^*_{kf}}{\partial X_{ig}}\frac{dX^*_{ig}}{d\Xi_{me}} + \frac{\partial\eta^*_{kf}}{\partial\Xi_{me}}\right)\right]$$
$$+ \frac{1}{\sigma^2_x}\left[\frac{dX^*_{ig}}{d\Xi_{nd}}\frac{dX^*_{ig}}{d\Xi_{me}}\right] + \frac{1}{\sigma^2_p}\delta_{nm}\delta_{de} \tag{12}$$

Plugging in the derivatives computed in the last section for each term above, we can approximate the Hessian above and compute the log Bayesian evidence by

$$\text{Log-Evidence} = -\left(\mathcal{L}(\Xi^*, \mathbf{X}^*) + \frac{N_\Xi}{2}(\log(2\pi) + 2\log\sigma_p) + \frac{1}{2}\log\left|\frac{1}{2\pi}\frac{d^2\mathcal{L}}{d\Xi d\Xi}\right|\right)$$

in which N_Ξ is the number of active terms in the model. Lastly, the Greedy algorithm is used to find the model that maximizes the log-evidence above.

A.3 Comparison with DAHSI

The main DAHSI algorithm generates many models from sweeping the thresholding (λ) and Variational Annealing (α^β) hyperparameters, from which a model is selected semi-automatically by information criterion or Pareto front. Given that certain manual inputs are anticipated in the model selection procedure, a direct comparison with ODR-BINDy on its capacity for automatic model recovery may not be the most appropriate. Nevertheless, as a highly generous metric, one can calculate the rate of identifying the true model in **at least one model** produced by the algorithm, which is equivalent to assuming that the subsequent model selection process via AIC/BIC or Pareto front will inevitably choose the correct model from the parameter sweep. Despite the favourable metric, fig. 11 shows that DAHSI cannot recover the original model most of the time in the noise regime examined by this work. Instead, DAHSI works only in a very low noise regime.

A.4 Comparison with SIDDs

Although we were unable to reproduce the results of SIDDs due to absent dependencies, we made every effort to compare our findings with the compiled plots shown in the original SIDDs publication. Upon approximating the result of Fig. 5.4 in [12] in fig. 11, which illustrates the success

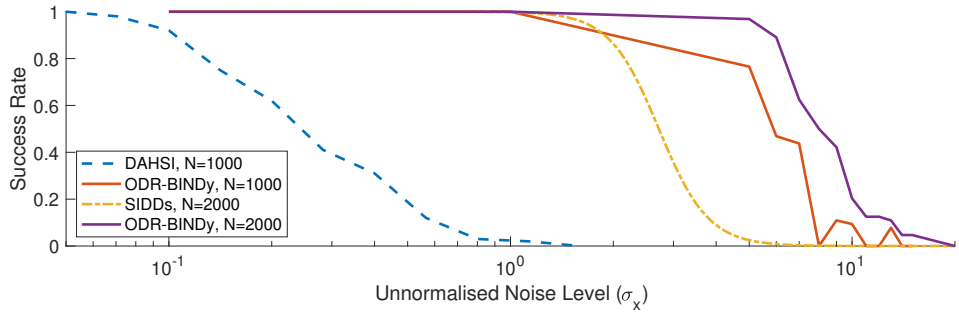


Fig. 11. Success rate of DAHSI, SIDDs and ODR-BINDy in recovering the Lorenz63 model from 1000/2000 data points, sampled at $\Delta t = 0.01$. Here, the hyperparameters used in DAHSI are $\alpha = 1.1$, $\beta_{\max} = 256$, $\lambda = 0.05 : 0.05 : 1.01$, whereas the data of SIDDs is a rough estimate from fig 5.4 of [12]. Data of ODR-BINDy are generated using the same settings as before. Note that the metric for "success" for DAHSI is more generous than the metric for ODR-BINDy. In all cases, ODR-BINDy is shown to be more data efficient and noise robust in recovering Lorenz63 from data than DAHSI and SIDDs.

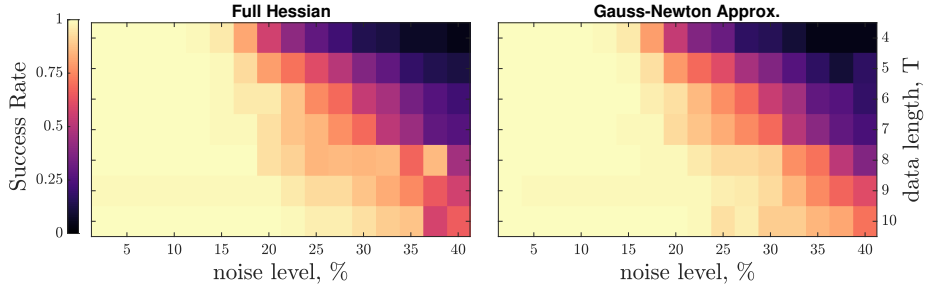


Fig. 12. Comparison of success rate of ODR-BINDy in recovering the Lorenz63 model sampled at $\Delta t = 0.01$ using the full computation of the Hessian (left) and the Gauss-Newton approxmation (right) for the Laplace Approximation of the Bayesian evidence, the metric used to score different models. The use of Gauss-Newton have little effect on the overall performance. The hyperparameters used here is the same as fig. 5.

rate in retrieving the same Lorenz63 equations using 2000 data points sampled at $\Delta t = 0.01$, we assert that ODR-BINDy demonstrates superior performance compared to SIDDs. This is likely attributable to the stochastic relaxation and Bayesian approach utilized in this study.

A.5 Gauss-Newton Approximation

Fig. 12 shows the performance comparison between employing the full evidence approximation (with Hessian) that the Gauss-Newton approximation of the evidence (without Hessian). The use of Gauss-Newton approximation seems to have little impact on the overall success rate of recovering the true model, but the performance gain of avoiding the full Hessian computation is substantial.

A.6 DySMHO `CONOPT` vs `IPOPT`

Due to licensing issue, we have only had a brief period of time where we have access to GAM's `CONOPT` that was used in the original work [17] . To work around the issue, we have replaced `CONOPT` with the open source `IPOPT` with `HSL`. To make sure the swap doesn't induce a significant performance change, we have compared the two methods. The comparison is shown in fig. 13, which shows that the replacement of `CONOPT` wtih `IPOPT` actually slightly improve overall performance.

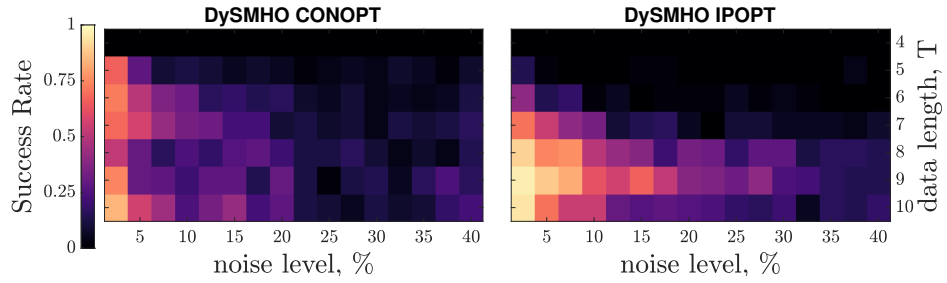


Fig. 13. Comparison of success rate of DySMHO in recovering the Lorenz63 model sampled at $\Delta t = 0.001$ using `CONOPT` (left) and `IPOPT` (right). Here, the hyperparameters used are: time window = 2 time unit, time step between window = 100, `nfe`=50, ncp=5, thresholding frequency =10 and thresholding tolerance=1. Notice that `IPOPT` is slightly better performing than `CONOPT`.

A.7 Hyperparameters of each algorithms

Tbl. 1 listed all the hyper-parameters used by each algorithm to learn from the examples listed in § 3. The success rate and model coefficient error statistics shown in figs. 4 to 6 and figs. 8 to 10 are computed by running each algorithm multiple times at each noise level and data length, each time with a different noise seed. While the author attempts to run these tests as many times as possible, owing to the computational cost of the endeavour, each algorithms are run different number of times: E-SINDy - 64 times, DySMHO - 64 times, modified-SINDy - 20 times and ODR-BINDy - 64 times.

Received 31 October 2025; revised 10 June 2026; accepted 10 July 2026

Example	Ensemble-SINDy	modified-SINDy	DySHMO	ODR-BINDy
Van Der Pol	λ: 0.2, nEnsembles: 500, ensembleT: 0.6, Weak Formulation: 9 pts, shape func: $\phi = (t^2-1)^2$	q: 1, λ: 0.05, ω: 0.9, Nloop:8, N_train:15000, N_SINDy_Iter: 15, NormalizeLib: false	nfe: 80, ncp: 15, data_step: 100, thresholding_frequency: 10, thresholding_tolerance: 1, horizon_length: 2.0, pre_processing_2 significance: 0.7	σ_x: measuremet noise level prescribed, $\sigma_{\partial t}$: 1e-2, σ_Y: 10, Finite Difference order: 4th
Lorenz63	λ: 0.2, nEnsembles: 100, ensembleT: 0.6, Weak Formulation: 7 pts, shape func: $\phi = (t^2-1)^2$	q: 3, λ: (0.1 if noise< 10%, 0.15 otherwise), ω: 0.9, Nloop:8, N_train:15000, N_SINDy_Iter: 15, NormalizeLib: false	nfe: 80, ncp: 5, data_step: 100, thresholding_frequency: 10, thresholding_tolerance: 1, horizon_length: 2.0, pre_processing_2 significance: 0.7	σ_x: measuremet noise level prescribed, $\sigma_{\partial t}$: 1e-3, σ_Y: 100, Finite Difference order: 6th
Rössler	λ: 0.05, nEnsembles: 500, ensembleT: 0.65, Weak Formulation: 7 pts, shape func: $\phi = (t^2-1)^2$	q: 1, λ: 0.05, ω: 0.9, Nloop:8, N_train:15000, N_SINDy_Iter: 15, NormalizeLib: false	nfe: 80, ncp: 5, data_step: 100, thresholding_frequency: 10, thresholding_tolerance: 1, horizon_length: 5.0, pre_processing_2 significance: 0.7	σ_x: measurement noise level prescribed, $\sigma_{\partial t}$: 5e-3, σ_Y: 50, Finite Difference order: 6th

Table 1. Hyperparameter settings used for each algorithms in the examples listed in § 3.