SINDy-SA: Enhancing Nonlinear System Identification with Sensitivity Analysis

Gustavo T. Naozuka (naozuka@lncc.br)
LNCC: Laboratorio Nacional de Computacao Cientifica  https://orcid.org/0000-0002-7331-9763

Heber L. Rocha
Indiana University Bloomington

Renato S. Silva
LNCC: Laboratorio Nacional de Computacao Cientifica

Regina C. Almeida
LNCC: Laboratorio Nacional de Computacao Cientifica

Research Article

Keywords: Sparse identification, Sensitivity analysis, Model selection, Differential equations, Data-driven methods

Posted Date: February 9th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1317964/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License. Read Full License
SINDy-SA: Enhancing nonlinear system identification with sensitivity analysis

Gustavo T. Naozuka · Heber L. Rocha · Renato S. Silva · Regina C. Almeida

Received: date / Accepted: date

Abstract Machine learning methods have revolutionized studies in several areas of knowledge, helping to understand and extract information from experimental data. Recently, these data-driven methods have also been used to discover structures of mathematical models. The sparse identification of nonlinear dynamics (SINDy) method has been proposed with the aim of identifying nonlinear dynamical systems, assuming that the equations have only a few important terms that govern the dynamics. By defining a library of possible terms, the SINDy approach solves a sparse regression problem by eliminating terms whose coefficients are smaller than a threshold. However, the choice of this threshold is decisive for the correct identification of the model structure. In this work, we build on the SINDy method by integrating it with a global sensitivity analysis (SA) technique that allows to classify terms according to their importance in relation to the desired quantity of interest. The proposed SINDy-SA approach thus eliminates the need to define the SINDy threshold. We compare our method with the original SINDy approach employing them in a variety of applications whose simulated data have different behaviors. For each application, we formulate different experimental settings and select the best model for both methods using model selection techniques based on information criteria. The results demonstrate that the SINDy-SA framework is a promising methodology to accurately identify interpretable data-driven models.

Keywords Sparse identification · Sensitivity analysis · Model selection · Differential equations · Data-driven methods

1 Introduction

Machine learning methods have been commonly used to understand behaviors, recognize patterns and make predictions from experimental data. Furthermore, another application of these methods, which has become popular in recent years, is the structural identification of mathematical models [2, 7, 6, 25, 20, 35, 37, 29]. These models, in turn, help to interpret the dynamics and allow the use of tools for mathematical analysis. Recently, Brunton et al. [2] have developed the sparse identification of nonlinear dynamics (SINDy) method, combining sparsity-promoting techniques and machine learning with nonlinear dynamical systems to discover governing equations from noisy measurement data. The only assumption about the structure of the mathematical model is that there are only a few important terms that govern the dynamics. Thus, by defining a prior set of possible functions, the authors solved a sparse regression problem with the objective of determining the smallest number of terms in this set, for each equation of the dynamical system, required to accurately represent the data. The SINDy method has been applied to solve many system identification problems, such as the dynamics of COVID-19 global transmission [17], the Duffing oscillator [23], and empirical data from measles, varicella, rubella, and chickenpox datasets [13, 12].

Several other related approaches have been published in the literature in order to recover the govern-
ing equations from data. França et al. [7] combined several machine learning techniques to improve the robustness to noise in system identification problems. Cortiella et al. [6] proposed an iterative sparse-regularized regression method to recover nonlinear dynamical systems from noisy measurement data. The work aims to improve the accuracy and robustness of SINDy in the presence of noisy state measurements, by developing a reweighted $\ell_1$-regularized least-squares solver. Niven et al. [26, 25] used an inverse Bayesian method for system identification from time-series data and compared two Bayesian methods, based on the joint maximum a posteriori and variational Bayesian approximation, to the SINDy approach. These methods are also used to quantify the variances of the model parameters. Maddu et al. [20] proposed a statistical learning framework based on group-sparse regression to discover mathematical models from data. This framework can enforce conservation laws, ensure model equivalence and symmetries, by using the group iterative hard thresholding algorithm and stability selection. Wang et al. [35, 36] presented a variational framework for system identification of partial differential equations, based on stepwise regression. The authors also addressed the influences of variable fidelity and noise in the measurement data. Yang et al. [37] presented a machine learning framework for Bayesian system identification from noisy, sparse, and irregular observation data. This framework uses differentiable programming and performs Bayesian inference using Hamiltonian Monte Carlo sampling. Rudy et al. [29] proposed a sparse regression method to extract partial differential equations from time-series measurements in the spatial domain. This method selects the nonlinear and partial derivative terms of the governing equations that best fit the data.

In addition, there are a number of extensions of the SINDy approach to improve its accuracy and robustness or to discover other types of mathematical models. Subramanian [33] incorporated a scientific machine learning approach in the context of clinical trials to discover governing equations of IPF (idiopathic pulmonary fibrosis) disease progression. This approach combines machine learning techniques, statistical methodologies, and scientific computing tools such as bootstrap sampling, cubic spline interpolation, Bayesian inference, and SINDy to discover the dynamics and quantify the uncertainty in the model parameters. Hoffmann et al. [11] extended the SINDy method to vector-valued ansatz functions, in order to estimate effective reaction networks from observations. In the proposed “reactive SINDy” approach, each function represents a particular reaction process. Boninsegni et al. [1] also extended the SINDy method to discover stochastic dynamical systems of biophysical processes. Jiang et al. [15] proposed a SINDy-LM modeling method, in which the SINDy approach is used to discover nonlinear dynamical systems from observation data and the Levenberg-Marquardt algorithm is used to improve the accuracy of the mathematical model identified by the SINDy algorithm. Hirsh [10] developed the theory of existing methods and proposed new techniques to model spatiotemporal data. These data-driven methods include dynamic mode decomposition (DMD), a dimensionality reduction method for time-varying linear dynamics, the Hankel Alternative View of Koopman (HAVOK) algorithm, and an uncertainty quantification for sparse identification of nonlinear dynamics (UQ-SINDy) method. The modified SINDy approach uses compressed sensing and Bayesian statistics to discover governing equations from data and quantifies model uncertainties. Brunton et al. [3] generalized the SINDy algorithm to identify nonlinear dynamical systems with external inputs and feedback control. Mangan et al. [21] proposed an alternative data-driven technique to identify networked nonlinear dynamical systems by using the SINDy algorithm. This technique can be used to recover governing equations that have rational function nonlinearities with cross terms. Kaheman et al. [16] also developed a variant of the SINDy algorithm to identify implicit dynamics and rational nonlinearities, by using multiple optimization algorithms and a model selection approach. This variant can be used to recover implicit differential equations and conservation laws from limited and noisy data. Quade et al. [28] proposed a conceptual framework to recover dynamical systems in response to abrupt changes from limited data. This framework first detects the abrupt change and then applies the SINDy method to update a previously identified mathematical model with the fewest changes.

Although the SINDy method is often able to correctly identify the structure of the mathematical model, this identification depends on the suitable choice of a threshold, which is used in the process of eliminating terms from the governing equations. One way to bypass the difficulty of this choice is to define a set of values for the threshold, run the method for each defined value, and select the best model from the resulting set of models, given the experimental data [2, 22]. However, the best value for the threshold may not belong to the defined prior set, resulting in incorrect identification of the dynamical system. Thus, to discover the most parsimonious model that best fits the data, one must run the SINDy method for a sufficiently large set of threshold values, which consequently increases the computational cost of solving the problem.
On the other hand, sensitivity analysis (SA) is a technique that allows classifying the parameters of a mathematical model according to their importance in relation to the desired quantity of interest [31, 5]. In our case, the quantity of interest is associated with the formulation of the sparse regression problem, described in detail in Subsection 2.1. In this work, we modify the implementation of the original SINDy method, replacing the need to determine the threshold with a global sensitivity analysis technique. In particular, we use the elementary effects method, a simple global SA approach, able to rank the model parameters according to their importance. We employ our method in a variety of applications using simulated data from a tumor growth model, a prey-predator model, a pendulum equation, and a compartmental model, and compare the obtained results with the original SINDy approach. For each application, we design different experimental settings, varying, for example, the set of possible functions for the governing equations. We then compare the models obtained against the experimental data in two ways: (i) constructing the Pareto curve that balances the accuracy and complexity of the models, (ii) and calculating the first- and second-order Akaïke and Bayesian information criteria, which weigh the goodness-of-fit and the number of model parameters. For the SINDy-SA results, the criteria used are able to correctly identify the best model, equivalent to the true model that generated the experimental data.

This paper is organized as follows. In Section 2, we explain some preliminary concepts required to understand the development of the proposed SINDy-SA method. Moreover, we describe in detail the SINDy-SA method and its implementation and show our general framework for solving system identification problems. In Section 3, we present the results obtained for the different applications by employing our method and the original SINDy approach. Finally, in Section 4, we point out some final remarks.

2 Problem Statement and the SINDy-SA Framework

Before introducing the new SINDy-SA method, we present in Subsection 2.1 some preliminary concepts about the sparse regression problem. Subsection 2.2 details our proposed SINDy-SA approach and its implementation. Finally, Subsection 2.3 explains our developed general framework for solving system identification problems.

2.1 Problem Statement

Consider that we are interested in determining the dynamics of a system of $n$ variables $x^{T} = \{x_1, x_2, \ldots, x_n\}, x(t) \in \mathbb{R}^n$. Under the assumption of spatial homogeneity, we assume that the rate of change of these variables is given by the following dynamical system:

\[ \frac{d}{dt} x(t) = f(x(t)), \]

in which the vector function $f(x(t))$ that we want to determine defines the interplay among the state variables. We approach the problem of determining the mathematical model (1) using sparse regression techniques, based on the fact that $f(x(t))$ has only a few terms that govern the dynamics of the system. The sparse regression technique is built upon the following available measured data that display a temporal history of the state variable vector $x(t)$ at multiple time instants $t_1, t_2, \ldots, t_m$:

\[ X = \begin{bmatrix} x_1^T(t_1) \\ x_2^T(t_1) \\ \vdots \\ x_1^T(t_m) \\ x_2^T(t_m) \end{bmatrix} = \begin{bmatrix} x_1(t_1) & x_2(t_1) & \cdots & x_n(t_1) \\ x_1(t_2) & x_2(t_2) & \cdots & x_n(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(t_m) & x_2(t_m) & \cdots & x_n(t_m) \end{bmatrix}. \]

From these data, we determine the corresponding temporal history of the state variable derivatives $\dot{x}(t)$, usually approximated numerically, which is likewise arranged in the form of an $m \times n$ matrix denoted by $\dot{X}$. With these definitions, the sparse regression problem aims to determine the $\ell \times n$ sparse matrix of coefficients $\Xi = [\xi_1, \xi_2, \ldots, \xi_\ell]$ so that

\[ \dot{X} = \Theta(X)\Xi, \]

in which $\Theta(X)$ is the $m \times \ell$ matrix of candidate nonlinear functions from the columns of $X$, and $\ell$ is the basis dimension for the function library $\Theta(X)$. This meaningful library includes candidate functions (polynomials, trigonometric functions, etc.) for the right-hand side of Equation (1), and is built from the modeler’s a priori knowledge of potential functions capable of describing the experimental data behavior.

The sparse regression problem (3) can be solved using classical regression methods such as least-squares, ridge, lasso, and elastic net [14, 8]. The original SINDy approach is implemented using the sequential thresholded least-squares (STLSQ) method [2]. Specifically, the STLSQ method requires the definition of a threshold for the values of the components of the sparse vectors $\xi_i$ below which the corresponding library functions are eliminated from the set of possible functions. The chosen threshold is decisive for the correct identification of the mathematical model. Thus, determining its best
value can be a hard task, especially when the dynamical system has parameter values with very different orders of magnitude. In such cases, additional approaches to overcome this difficulty are required [2, 22]. For more information, we refer the reader to the Supplementary Material (Section SM-1). After determining the sparse matrix \( \Xi \), the desired identified system is obtained as

\[
\dot{x} = f(x) = \Xi^T (\Theta(x^T))^T.
\]  

2.2 The Proposed SINDy-SA Approach

The original SINDy method depends on the proper choice of the threshold, necessary to eliminate candidate functions from the governing equations. Due to the difficulty of this choice and the consequent increase in computational cost to determine its best value, as suggested in Brunton et al. [2], we propose in this work to integrate the sparse regression problem with a global SA technique. By using a SA technique that is able to rank the coefficient terms according to their importance with respect to \( \dot{X} \), the selected quantities of interest (QoIs), we overcome the need of using a pre-defined threshold to eliminate candidate functions. Instead, the non-influential terms can be eliminated and the accuracy of the regression verified. In particular, we use the Morris method, also known as the Elementary Effects (EE) method, due to its simplicity and low computational cost [31]. Moreover, the EE method assesses not only the overall importance of the coefficients but also their interactions and nonlinear effects.

The implementation of the proposed approach was carried out in the Python programming language, based on the PySINDy module, a sparse regression package containing different implementations for the SINDy method [32]. The EE method was developed using the SALib module, composed of commonly used SA methods [9].

Our proposed SINDy-SA sparse regression method is an iterative process encompassing the ridge regression estimation, the error computation, and the SA technique. At every \( \tau \)-th iteration, non-influential terms are eliminated until only influential terms remain. Algorithm 1 presents a pseudocode of the SINDy-SA method for solving the sparse regression problem (3), in which \( \hat{\Xi} \) describes the estimate for the sparse vectors of coefficients in the iteration \( \tau \). Additionally, Figure SM-1 in the Supplementary Material illustrates a flowchart of our method, graphically representing each instruction of Algorithm 1. We detail the three main features of our SINDy-SA method in the following.

\begin{algorithm}
\caption{SINDy-SA}
1: \( \tau \leftarrow 0; \)
2: \textbf{repeat}
3: \hspace{1em} Compute a least-squares solution with \( \ell_2 \)-regularization for \( \Xi \); 
4: \hspace{1em} Find the derivative \( \dot{\hat{X}}_\tau \) using \( \hat{\Xi} \); 
5: \hspace{1em} Calculate the SSE between the derivatives \( \dot{\hat{X}} \) and \( \dot{\hat{X}}_\tau \); 
6: \hspace{1em} if \( \tau > 0 \) and \( \text{SSE}(\dot{\hat{X}}, \dot{\hat{X}}_\tau) > M_\tau + \varepsilon D_\tau \) \text{ then} 
7: \hspace{2em} return \( \hat{\Xi}_{\tau - 1} \); 
8: \hspace{1em} end if 
9: \hspace{1em} Perform sensitivity analysis of model parameters; 
10: \hspace{1em} if all parameters are important \text{ then} 
11: \hspace{2em} return \( \hat{\Xi}_\tau \); 
12: \hspace{1em} else 
13: \hspace{2em} Eliminate terms with less importance in relation to the QoI; 
14: \hspace{2em} end if 
15: \hspace{1em} \( \tau \leftarrow \tau + 1 \); 
16: \textbf{end repeat}
\end{algorithm}

\textbf{Ridge Regression}

In our SINDy-SA approach, we start by assigning to each column \( \hat{\xi}_\tau \) of \( \hat{\Xi} \) the corresponding least-squares solution with \( \ell_2 \)-regularization. Specifically, defining \( \hat{\hat{X}}_k \) as the \( k \)-th column of \( \hat{\hat{X}} \), we use the ridge regression [14, 8]:

\[
\hat{\xi}_\tau = \arg \min_{\xi_k} \| \hat{\hat{X}}_k - \Theta(\dot{\hat{X}})\xi_k \|_2^2 - \alpha \| \xi_k \|_2^2 
\]

(5)

to estimate the coefficient values, balancing two different criteria: the residual sum of squares and a penalty term that has the effect of preventing overfitting. The tuning parameter \( \alpha \) serves to control the relative impact of these two terms on the coefficient estimates. Different \( \alpha \) values are likely to lead to different coefficient estimates and, ultimately, to different models. Of note, we choose this regression method to estimate the sparse vector of coefficients based on the fact that the PySINDy module also implements it and several previous experiments in which we investigated the accuracy of commonly used statistical regression methods to identify nonlinear dynamical systems.

\textbf{Error Computation}

After the ridge regression estimation, we then calculate the \( \tau \)-th estimate for the derivative \( \dot{\hat{X}}_\tau \) based on the library \( \Theta(X) \) and the estimates for the sparse vectors of coefficients \( \hat{\Xi}_\tau \) obtained by the ridge regression, according to the following equation:

\[
\dot{\hat{X}}_\tau = \Theta(X) \hat{\Xi}_\tau.
\]  

(6)

Next, we calculate the sum of squared errors (SSE) between the derivatives \( \dot{\hat{X}} \), measured or approximated...
from the data, and \( \hat{X}_\tau \), given by Equation (6), through:

\[
\text{SSE}_\tau(\hat{X}, \hat{X}_\tau) = \sum_{k=1}^{n} \sum_{i=1}^{m} \left( \hat{x}_k(t_i) - \hat{x}_{k,i}(t_i) \right)^2, \tag{7}
\]

where \( \hat{x}_k(t_i) \in \hat{X} \) and \( \hat{x}_{k,i}(t_i) \in \hat{X}_\tau \). We store the SSE values for each iteration of the SINDy-SA method in order to determine whether the error between the current and previous iterations has significantly increased. In the first iteration (\( \tau = 0 \)), no error comparison is performed. The iterative process continues until a significant increase in the SSE occurs, ending the process. Defining what is significant is a tricky task since SSE fluctuations can occur depending on the problem. To overcome this issue, we define an iteration window, denoted by \( W \), in which we collect statistics from part of the previous evolution of the SSE, which are used as a reference. In mathematical terms, we check the condition:

\[
\text{SSE}_\tau(\hat{X}, \hat{X}_\tau) > M_\tau + \varepsilon D_\tau, \tag{8}
\]

where \( M_\tau \) and \( D_\tau \) represent, respectively, the mean and the standard deviation of the errors of previous iterations belonging to \( W \), and \( \varepsilon \) is a scaling factor. Some adjustments are required when \( \tau < W \) and are given according to the conditions described in Table 1.

Note that, for the second iteration of the algorithm (\( \tau = 1 \)), the standard deviation \( D_1 \) would be zero considering only the error of the previous iteration. For this reason, we consider \( D_1 \) as 10% of the SSE obtained in the first iteration. The values of the parameters \( W \) and \( \varepsilon \) can be determined by observing the sequence of SSE values, numerically or graphically, and the goodness-of-fit of the resulting mathematical model simulation against the experimental data. In all our experiments, we set the window size \( W = 3 \) iterations, while the parameter \( \varepsilon \) varies for each experiment.

If the condition (8) is satisfied, the algorithm ends, and the coefficient estimates of the iteration \( \tau - 1 \), whose error was considered significantly smaller compared to the iteration \( \tau \), are used for the construction of governing equations. Otherwise, the procedure continues and performs the SA of model parameters.

**Sensitivity Analysis**

In the SA of model parameters, we firstly determine admissible variations for the non-zero coefficients, being \( \pm 20\% \) of the estimated value for the coefficient. Given one or more analysis time instants, we calculate the sensitivity indices \( \mu_{k,i} \) and \( \sigma_{k,i} \), that indicate the influence and nonlinear importance of the \( i \)-th parameter on the \( k \)-th QoI, respectively. The higher the values of these sensitivity indices, the more influential are the corresponding parameters, so that small variations in their values have a great impact on the estimation of \( \hat{X}_\tau \). For more details about the EE method, see Supplementary Material (Section SM-2).

In order to facilitate sorting parameters by their order of importance with respect to QoI and to take into account both their direct and nonlinear effects, we combine the sensitivity indices into a single metric:

\[
S_{k,i} = \sqrt{(\mu_{k,i})^2 + (\sigma_{k,i})^2}. \tag{9}
\]

In this way, the greater the value of \( S_{k,i} \), the greater is the overall influence of the \( i \)-th parameter on the \( k \)-th QoI. For each equation of the dynamical system, we order the sensitivity indices \( S_{k,i} \) from lowest to highest, we eliminate the least important terms, ensuring that the influential terms are kept. Thus, each governing equation must contain at least one active candidate function at the end of the entire process, assuming that the modeler knows a priori the number of state variables in the system.

We develop a scoring scheme to integrate SA at multiple time instants. Each model parameter receives a score \( P_{k,i} \), depending on its rank of importance which is updated over the analysis times, starting from 0. Of note, \( k = 1, \ldots, n \) and \( i = 1, \ldots, d \), where \( n \) is the number of state variables, and \( d \) is the number of model parameters. Once, at a given time of analysis, the model parameters are ordered from the least important parameter \( \hat{\xi}_{k,1} \) to the most important parameter \( \hat{\xi}_{k,d} \), we ensure that the latter terms are kept by assigning \( -1 \) to \( P_{k,d} \). Next, we increment an increasing score \( \pi \) to the total score \( P_{k,i} \) for the remaining parameters of the model. In this way, more influential parameters receive a lower score, and parameters with less influence receive a higher score. After analyzing all time instants, we check if \( P_{k,i} = -1 \) for all \( k = 1, \ldots, n \) and \( i = 1, \ldots, d \), meaning that all parameters are important. Otherwise, we eliminate the terms that have the highest total score \( P_{k,i} \), being able to eliminate more than one candidate function for each state variable. Algorithm 2 presents the pseudocode of the procedure for sensitivity analysis at multiple time instants.

If all parameters are considered important, the algorithm ends, in which case the coefficient estimates of the iteration \( \tau \) are used to construct the governing equations. Otherwise, the procedure returns to the first step calculating a least-squares solution with \( \ell_2 \)-regularization for \( \hat{X}_\tau \). It is important to observe that, in our approach, it is not necessary to specify a maximum number of iterations because eventually or all the
Table 1: Mathematical definitions for the mean $\mathcal{M}_\tau$ and the standard deviation $\mathcal{D}_\tau$ for each iteration of the SINDy-SA method, according to three conditions: $\tau = 1$, $1 < \tau < W$, and $\tau \geq W$. $W$ denotes the window size of previous iterations used to check the condition (8).

<table>
<thead>
<tr>
<th>$\tau = 1$</th>
<th>$1 &lt; \tau &lt; W$</th>
<th>$\tau \geq W$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{M}_\tau$</td>
<td>$\frac{1}{\tau} \sum_{i=0}^{\tau-1} \text{SSE}_i$</td>
<td>$\frac{1}{W} \sum_{i=\tau-W}^{\tau-1} \text{SSE}_i$</td>
</tr>
<tr>
<td>$\mathcal{D}_\tau$</td>
<td>$0.1 \text{SSE}_0$</td>
<td>$\sqrt{\frac{1}{\tau} \sum_{i=0}^{\tau-1} (\text{SSE}<em>i - \mathcal{M}</em>\tau)^2}$</td>
</tr>
</tbody>
</table>

Algorithm 2 Sensitivity analysis at multiple time points

1: Initialize the total score $P_{k,i} \leftarrow 0$, with $k = 1, \ldots , n$ and $i = 1, \ldots , d$;
2: for each time instant for analysis do
3: for each state variable $k = 1, \ldots , n$ do
4: Order the parameters associated with the state variable $k$ according to their combined sensitivity index $S_{k,i};$
5: $P_{k,d} \leftarrow -1$, where $\xi_{k,d}$ is the most important parameter in relation to the QoI;
6: $\pi \leftarrow 1$;
7: for each parameter $\xi_{k,i}$, with $i = d - 1, \ldots , 1$ do
8: if $P_{k,i} \neq -1$ then
9: $P_{k,i} \leftarrow P_{k,i} + \pi$;
10: $\pi \leftarrow \pi + 1$;
11: end if
12: end for
13: end for
14: end for
15: if $P_{k,i} = -1$, for all $k = 1, \ldots , n$ and $i = 1, \ldots , d$ then
16: All parameters are important;
17: else
18: return the terms with the highest total score $P_{k,i}$ (i.e., the least importance in relation to the QoI);
19: end if

model parameters are considered important or condition (8) is satisfied. Indeed, if a model parameter is erroneously eliminated according to SA, the SSE between the current and previous iterations tends to increase, and the condition (8) must be satisfied.

2.3 General Framework for System Identification

The proposed framework for solving the problem of identifying nonlinear dynamical systems is represented schematically in Figure 1. Given the experimental dataset $X$ and a library $\Theta(X)$ of candidate nonlinear functions, we define a set of $N$ experiments, in which we vary the library $\Theta(X)$ and the regularization parameter $\alpha$ of the ridge regression. In this way, we can investigate the influences of choosing different potential configurations in modeling.

For each experiment, we run the SINDy-SA method to solve the sparse regression problem, consisting basically of the three steps described in Subsection 2.2: the ridge regression for estimating the sparse vectors of coefficients $\hat{\xi}_\tau$, the computation of SSE between the derivatives $\dot{X}$ and $\dot{X}_\tau$, and the sensitivity analysis to eliminate less important terms from the governing equations. Alternatively, in our general framework for system identification, the SINDy-SA method can be replaced with other approaches for identifying nonlinear dynamical systems [2, 7, 6, 25, 20, 35, 37, 29].

Once the mathematical model is identified, we can optionally improve the accuracy of the estimate for the sparse vectors of coefficients $\hat{\xi}_\tau$, using, for example, the Levenberg-Marquardt optimization algorithm, as suggested in Jiang et al. [15], or the Bayesian calibration, through which parameter uncertainties can be quantified. The model recalibration step is optional, but it can be critical for selecting the best model in relation to the experimental data.

After applying the SINDy-SA method and recalibrating the model, the set of experiments yields a set of candidate models from which the best model is selected using a model selection technique. Here, we use some of the most used model selection methods for comparing both nested and non-nested models: the Akaike Information Criterion (AIC), the second-order Akaike (AICc), and the Bayesian Information Criterion (BIC) [4], which are described in the Supplementary Material (Section SM-3) for completeness.

3 Applications and Results

In this section, we employ the proposed SINDy-SA approach in a variety of applications with different behaviors. We also employ the original SINDy method in the same applications to compare the results. We use simulated data from a prey-predator model (Subsection 3.1), a tumor growth model (Subsection 3.2), a pendulum equation (Subsection 3.3), and a compartmental model (Subsection 3.4). Specifically for the prey-
SINDy-SA: Enhancing nonlinear system identification with sensitivity analysis

Fig. 1: Schematic representation of the framework for solving the problem of identifying nonlinear dynamical systems, using the proposed SINDy-SA approach. The flowchart detailing the SINDy-SA method is shown in the Supplementary Material.

Predator model, we detail all steps of Algorithm 1 in order to improve the understanding of the proposed approach. In Subsection 3.5, we apply the SINDy-SA method to limited and noise-contaminated data scenarios and briefly present the limitations of our proposed approach.

Importantly, we use the LSODA (Livermore Solver for Ordinary Diﬀerential Equations) method [27] to numerically simulate the mathematical models and collect measurement data. We approximate the derivative \( \dot{X} \) from the collected data using the second-order ﬁnite diﬀerence method. For each application, we elaborate diﬀerent experimental settings, varying the candidate nonlinear function library \( \Theta(X) \), the tuning parameter \( \alpha \) of the ridge regression, and, for the SINDy experiments, the threshold \( \lambda \) of the STLSQ method. For the SINDy-SA experiments, we run the Levenberg-Marquardt optimization algorithm to enhance the accuracy of the coeﬃcient estimates. We then compare the resulting models against the available data by observing the Pareto curve and calculating the AIC, AICc, and BIC, and their corresponding weights that represent the probability of being the best model given the data.

3.1 Prey-Predator Model

In our ﬁrst application, we use the prey-predator model to generate the simulated data that will be used in the sparse identiﬁcation problem. This model describes the evolution of interactions between prey and predator over time \( t \), until \( t = t_f = 20.0 \), and is given by:

\[
\begin{align*}
\frac{dx}{dt} &= c_0 x + c_1 x y, \\
\frac{dy}{dt} &= c_2 y + c_3 x y, \\
x(0) &= x_0, \quad y(0) = y_0,
\end{align*}
\]

(10)

where \( x(t) \) and \( y(t) \) denote prey and predator populations, respectively. In this illustrative example, we use the prey birth rate \( c_0 = 1.0 \), the predator death rate \( c_2 = -1.0 \), and the interaction coefficients \( c_1 = -1.0 \) and \( c_3 = 1.0 \), and the initial conditions are set equal to \( (x_0, y_0) = (4.0, 1.0) \). The training set is built by simulating the model and collecting the numerical solutions at \( m = 200 \) equally distributed time points. Using these collected solutions, we approximate the derivative \( \dot{X} \) and build a library \( \Theta(X) \) consisting only of polynomials, varying their maximum degree \( P_i \), \( i \in \{2, 3, 4, 5\} \). For example, if the maximum degree of the polynomial is \( P_2 \) with two state variables representing prey and predator populations, there are six candidate functions.
so that

\[ \Theta(X) = \begin{bmatrix} 1 & x & y & x^2 & xy & y^2 \end{bmatrix} \]

is the 200 \times 6 matrix of potential candidate functions. For each chosen maximum degree, we also vary the tuning parameter \( \alpha \in \{10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1\} \) of the ridge regression, totaling 20 experiments for the SINDy-SA method. For the original SINDy approach, we additionally vary the threshold \( \lambda \in \{10^{-1}, 10^0, 10^1\} \) of the STLSQ method, ending up with 60 experiments.

In all SINDy-SA experiments, we use \( \varepsilon = 100.0 \) and perform the SA at \( t = t_f \).

To detail the steps of Algorithm 1, consider the experiment in which the maximum degree of the polynomial is \( P_2 \) and the tuning parameter is \( \alpha = 10^{-3} \). With these settings, the SINDy-SA method leads to the sequence of \( \tilde{E} \), SSE(\( \tilde{X}, \tilde{X} \)), \( M_\varepsilon \), and \( \varepsilon D \) displayed in Table 2 for each iteration. Figure 2 graphically relates the SSE(\( \tilde{X}, \tilde{X} \)) with the sum \( M_\varepsilon + \varepsilon D \), where the bar graphs represent the SSE, the points denote the mean \( M_\varepsilon \), and the vertical lines describe the margin of error \( \varepsilon D \). Figure 3 displays the heatmap of the total scores \( P_{k,i} \) calculated for all candidate functions in each iteration of the method. The corresponding combined sensitivity index \( S_{k,i} \) are indicated inside each heatmap cell. The total score \( P_{k,i} \) depicts four possible situations:

(i) \( P_{k,i} = -1 \): the term is considered to be more important in relation to QoI by SA and thus should be kept in the governing equations;
(ii) \( P_{k,i} = 0 \): the term was eliminated in some previous iteration;
(iii) \( P_{k,i} = 1, \ldots, 4 \): the term has medium importance and can be deleted or maintained; and
(iv) \( P_{k,i} = 5 \): the term is considered to be less important in relation to QoI by SA and thus should be eliminated from the governing equations.

According to Table 2 and Figures 2 and 3, the method performed six iterations, in each eliminating two candidate functions from the mathematical model. SSE increased as terms are eliminated, remaining relatively small until \( \tau = 4 \). In this iteration, the elimination of the nonlinear terms leads to a significant increase in the SSE of the iteration \( \tau = 5 \) that satisfies the stopping criterion (8). Thus, the identified model is the one associated with iteration \( \tau = 4 \) (step 7 of Algorithm 1), which corresponds to the true model although with parameters slightly different from 1.0. Those estimates are improved using the Levenberg-Marquardt optimization algorithm through which model (10) is recovered. It is important to note that, throughout the algorithm, each eliminated candidate function is associated with a state variable of the dynamical system. This behavior is due to the fact that we performed the SA of the parameters only at the final time of the experimental time frame for this illustrative example. If SA is performed at multiple time instants, each iteration can eliminate varying numbers of candidate functions. For a generic application, in which the true model is not known \textit{a priori}, we recommend that SA be performed at multiple time instants.

Running the SINDy-SA method for all 20 experiments resulted in four different mathematical models. For comparison, running the original SINDy method for all 60 experiments resulted in eight different mathematical models. In order to select the best model among the resulting set of models, we numerically simulate each dynamical system, given the initial conditions and the discretized time domain defined in Equation (10), and compare with the measurement data calculating SSE(\( \tilde{X}, \tilde{X} \)). Using \( m = 200 \) data, we calculate AIC, AICc, and BIC, and their corresponding weights, as shown in Table 3. Of note, Figure SM-2 in the Supplementary Material illustrates the Pareto curve relating the SSE to the model complexity, measured in terms of the number of parameters \( d \).

According to Table 3, the first model is the best according to all the three considered information criteria for the SINDy-SA method. This is due to the fact that the number of parameters in this model is relatively small, and the SSE(\( \tilde{X}, \tilde{X} \)) is significantly lower compared to the other identified models. On the other hand, the model selected by the information criteria for the original SINDy approach structurally corresponds to the true model (10), but its parameter values are approximately equal to 1.0 in absolute values. Figure 4 shows a comparison between the observed data and the numerical solution of the selected model for both SINDy-SA and SINDy approaches, as well as the corresponding identified best models. The best model in the original SINDy approach needs to be recalibrated using, for example, the Levenberg-Marquardt algorithm in order to improve its accuracy in representing the data. Therefore, the results obtained for this application demonstrate that the model recalibration step can be essential for the accurate identification of the true mathematical model. Specifically in this application, the model identified by the original SINDy method would be the same obtained by the SINDy-SA (true model) if there was a recalibration after the identification of the model. As we will see in the following, this is not always the case.
Fig. 2: SSE between the derivatives $\dot{\mathbf{X}}$ and $\hat{\mathbf{X}}$, mean $\mathcal{M}$ and margin of error $\epsilon \mathcal{D}$ at each iteration of the SINDy-SA method, considering the iteration window $W = 3$ and the scaling factor $\epsilon = 100.0$.

Table 3: Model selection results for the prey-predator application. Subscripts $w_i$ indicate the model selection criterion weights. Models selected as the best ones are indicated in the gray line.
3.2 Tumor Growth Model

In our next application, we consider the logistic growth model, previously calibrated using 14 tumor volume (in $\text{mm}^3$) data points from the breast cancer growth (GI-101A xenografts) in athymic mice, distributed over 114 days [24]. By denoting $x(t)$ as the tumor volume at time $t$, the evolution of the tumor volume over time, $0 < t \leq t_f = 300$ days, is then given by:

$$\begin{align*}
\frac{dx}{dt} &= c_0 x + c_1 x^2, \\
x(0) &= x_0,
\end{align*}$$

where $c_0 = 0.028$/day, $c_1 = -3.305 \times 10^{-6}$/($\text{day} \times \text{mm}^3$), and $x_0 = 245.185$ mm$^3$. The absolute value of $c_1$ is obtained by dividing the growth rate $c_0$ by the carrying capacity 8472.914 mm$^3$. The derivatives $\dot{X}$ are approximated from the numerical solution obtained by simulating model (11) at $m = 300$ evenly distributed time points. As in the previous application, we define 20 experiments for the SINDy-SA method by building $\Theta(X)$ using polynomials of degree at most equal to $P_i$, $i \in \{2, 3, 4, 5\}$, and varying $\alpha \in \{10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1\}$. We also vary the threshold $\lambda \in \{10^{-7}, 10^{-6}, 10^{-5}\}$ of the STLSQ method, which yields 60 experiments for the original SINDy approach. We use $\varepsilon = 10.0$ and we perform the SA at $t = t_f$.

The execution of the SINDy-SA method for all 20 experiments resulted in only two different mathematical models. For comparison, the execution of the original SINDy method for all 60 experiments resulted in three different mathematical models. Table 4 presents the sets of models discovered by each system identification method. The model comparison of the identified models is performed, and the results are displayed in Table 5. For completeness, Figure SM-3 in the Supplementary Material illustrates the corresponding Pareto curves.

All three model selection criteria agreed in choosing the best models, as shown in Table 5. While the SINDy-SA method identified the correct model, notice that the SINDy approach could not capture the correct structure of the model since it wrongly incorporates a source term. Figure 5 illustrates a comparison between the observed data and the numerical solution of the selected best model for each sparse identification method as well as the corresponding mathematical description. Thus, despite the SINDy-SA method having identified two mathematical models considering different experimental settings, the information criteria correctly selected the most parsimonious model that best fits the simulated data. In comparison, the original SINDy approach fails to solve the system identification problem in this example, mainly due to the difference between the orders of magnitude of the two terms in the true model. Therefore, the results obtained for this application show a limitation of using a threshold to eliminate candidate functions from the governing equations and highlight the importance of performing the SA of model parameters as in our proposed approach, allowing us to...
Fig. 4: Comparison between the observed data $\mathbf{X}$ and the numerical solution $\hat{\mathbf{X}}$ of the best predator-prey models selected by the information criteria, shown in the bottom right. The observed data are simulated from the prey-predator model (10), which corresponds to the best mathematical model identified by the SINDy-SA method. The best model identified by the original SINDy approach structurally corresponds to the prey-predator model (10), but its parameter values are not exactly equal to 1.0 in absolute values.

Table 4: Mathematical models identified by the SINDy-SA and SINDy methods after running all experiments using simulated data from the logistic model (11).

<table>
<thead>
<tr>
<th>Method</th>
<th>Model</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>SINDy-SA</td>
<td>1</td>
<td>$\dot{x} = 0.028x - 3.305 \times 10^{-6}x^2$</td>
</tr>
<tr>
<td>SINDy-SA</td>
<td>2</td>
<td>$\dot{x} = 0.028x - 3.305 \times 10^{-6}x^2 - 2.393 \times 10^{-16}x^3$</td>
</tr>
<tr>
<td>SINDy</td>
<td>1</td>
<td>$\dot{x} = 0.002 + 0.028x - 3.304 \times 10^{-6}x^2$</td>
</tr>
<tr>
<td>SINDy</td>
<td>2</td>
<td>$\dot{x} = 35.436 - 0.002x$</td>
</tr>
<tr>
<td>SINDy</td>
<td>3</td>
<td>$\dot{x} = 0.004x$</td>
</tr>
</tbody>
</table>

correctly discover the governing equations that generated the data.

3.3 Pendulum Motion Model

In this subsection, we apply the pendulum equation to generate the simulated data that is used in the dynamical system identification problem. The pendulum motion over time $t$ under the influence of gravity and fric-
Table 5: Model selection results for the tumor growth application. Subscripts $w_i$ indicate the model selection criterion weights. Models selected as the best ones are indicated in the gray line.

<table>
<thead>
<tr>
<th>Method</th>
<th>Model</th>
<th>$d$</th>
<th>SSE$(\mathbf{X}, \hat{\mathbf{X}})$</th>
<th>AIC</th>
<th>AIC$_c$</th>
<th>BIC</th>
<th>AIC $w_i$</th>
<th>AIC$_c$ $w_i$</th>
<th>BIC $w_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>SINDy-SA</td>
<td>1</td>
<td>2</td>
<td>$8.103 \times 10^{-17}$</td>
<td>$-12822.639$</td>
<td>$-12822.598$</td>
<td>$-12815.231$</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>3</td>
<td>$1.842 \times 10^{-5}$</td>
<td>$-4975.767$</td>
<td>$-4975.686$</td>
<td>$-4964.656$</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>SINDy</td>
<td>1</td>
<td>3</td>
<td>6.221</td>
<td>$-1156.734$</td>
<td>$-1156.652$</td>
<td>$-1145.622$</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2</td>
<td>227667433.107</td>
<td>4065.884</td>
<td>4065.925</td>
<td>4073.292</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1</td>
<td>8557132954.600</td>
<td>5151.875</td>
<td>5151.888</td>
<td>5155.578</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

(a) SINDy-SA method

\[
\dot{x} = 0.028x - 3.305 \times 10^{-6}x^2
\]

(b) SINDy method

\[
\dot{x} = 0.002 + 0.028x - 3.304 \times 10^{-6}x^2
\]

Fig. 5: Comparison between the observed data $\mathbf{X}$ and the numerical solution $\hat{\mathbf{X}}$ of the best tumor growth models selected by the used information criteria, which are shown inside the graphs. Although the dynamics are quite similar, the model in (b) includes a source term (shown in gray) that is not present in the true model.

The motion may be described in terms of the angle $x(t)$ and the angular velocity $y(t)$ through the following system of differential equations [34]:

\[
\begin{align*}
\frac{dx}{dt} &= 1.0y, \\
\frac{dy}{dt} &= c_0 y + c_1 \sin(x),
\end{align*}
\]

$\begin{align*}
x(0) &= x_0, & y(0) &= y_0.
\end{align*}$
In this example, we use the parameters $c_0 \approx -0.25$ and $c_1 \approx -0.5$, and we set $(x_0, y_0) = (\pi - 0.1, 0.0)$. System (12) is integrated up to time $t_f = 30.0$, and we collect the numerical solutions at $m = 301$ evenly distributed time points, which are used to approximate the derivative $\dot{X}$. Due to the specific dynamics observed in this problem, we build the library $\Theta(X)$ using both polynomials, varying their maximum degree $P_i$, $i \in \{1, 2, 3\}$, and trigonometric functions (sine and cosine), varying their frequency $F_j$, $j \in \{1, 2, 3\}$. If, for example, the maximum degree of the polynomial is $P_1$ and the frequency of the trigonometric functions is $F_1$, the library $\Theta(X)$ is given by:

$$
\Theta(X) = \begin{bmatrix}
1 & x & y & \sin(x) & \cos(x) & \sin(y) & \cos(y)
\end{bmatrix},
$$

which contains seven candidate functions. As in the previous examples, for each chosen maximum degree and frequency, we also vary $\alpha \in \{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}$, which means that we have 45 experiments for the SINDy-SA method. These experiments are also combined with possible variations of the threshold $\lambda$ for the original SINDy approach, leading to 135 experiments by setting $\lambda \in \{10^{-3}, 10^{-2}, 10^{-1}\}$. In all SINDy-SA experiments, we use $\varepsilon = 10^6$ and we perform the SA at three time instants over the experimental time frame, $t = 10.0, 20.0, 30.0$, to keep track of the dynamics of the pendulum motion.

The SINDy-SA and SINDy methods identified four and 84 different mathematical models, respectively. Table 6 details the model selection results and indicates the best model among the candidate set. To complement these results, Figure SM-4 in the Supplementary Material illustrates the corresponding Pareto curves relating the SSE with the model complexity.

As shown in Table 6, the first pendulum motion model was selected as the best among the four models identified by the SINDy-SA method for the three information criteria with weights of 100%. Regarding the original SINDy approach, the model selected by all information criteria does not correspond to the pendulum motion equation (12). Indeed, it incorporates three additional terms in the time evolution of the angle $x(t)$ and five terms in the time evolution of the angular velocity $y(t)$ that are not present in the true model. The AIC and AICc weights indicate that the probability of model 4 being the best model is around 91%, although the structure of the true model is present in the set of resulting models (model 3), which has approximately zero probability of being the best model. This happens due to inaccuracy of the coefficient values determined by the SINDy method so that all information criteria favored the best fit model. Figure 6 displays a comparison between the experimental data and the numerical solution of these best models for each sparse identification method, including their mathematical description.

In this application, the information criteria were able to correctly select the most parsimonious model that best fits the observed data for the SINDy-SA experiments. In comparison, the models identified by the original SINDy approach need to be recalibrated so that the true model can be selected by the information criteria. Therefore, the results obtained for this application also demonstrate that the model recalibration step can be important for the correct identification of the true mathematical model. Unlike the application of the prey-predator model, all discovered models must be recalibrated in this case since the true model structure was not selected by the information criteria, but it is present in the set of resulting models.

3.4 Compartmental Model

Our fourth application involves using the epidemiological SIR compartmental model to generate the simulated data as input to both SINDy-SA and SINDy methods. This model describes the evolution of a fixed population of $N$ individuals over time $t$ divided into three compartments: susceptible $S(t)$, infected $I(t)$, and recovered $R(t)$. For $t \leq t_f$, it is represented by:

$$
\begin{align*}
\frac{dS}{dt} &= c_0 SI, \\
\frac{dI}{dt} &= c_1 SI + c_2 I, \\
\frac{dR}{dt} &= c_3 I, \\
S(0) &= S_0, \quad I(0) = I_0, \quad R(0) = R_0, \\
N &= S(t) + I(t) + R(t).
\end{align*}
$$

By construction, $c_1$ are the transition coefficients that satisfy $c_0 = -c_1$ and $c_2 = -c_3$, so that $R(t)$ is known from the infected individuals $I(t)$. In this way, we can simplify the sparse identification problem of this model by removing the equation associated with the recovered individuals. To generate the training data, we collect the numerical solution of $m = 400$ evenly distributed time points using $t_f = 40.0$, the infection coefficient $c_0 = -c_1 = -4 \times 10^{-4}$ and the recovery rate $c_2 = -0.04$. We set $S_0 = 997$ and $I_0 = 3$ so that $N = 1000$ individuals ($R_0 = 0$). From the collected solution, we approximate the derivative $\dot{X}$ and build the SINDy-SA experiments by setting $\Theta(X)$ consisting of polynomials, varying their maximum degree $P_i$, $i \in \{2, 3, 4, 5\}$,
Table 6: Model selection results for the pendulum application. Subscripts \( w_i \) indicate the model selection criterion weights. Models selected as the best ones are indicated in the gray line.

<table>
<thead>
<tr>
<th>Method</th>
<th>Model</th>
<th>( d )</th>
<th>( \text{SSE}(\mathbf{X}, \mathbf{\hat{X}}) )</th>
<th>AIC</th>
<th>AICc</th>
<th>BIC</th>
<th>AIC ( w_i )</th>
<th>AICc ( w_i )</th>
<th>BIC ( w_i )</th>
</tr>
</thead>
<tbody>
<tr>
<td>SINDy-SA</td>
<td>1</td>
<td>3</td>
<td>( 6.554 \times 10^{-7} )</td>
<td>-5997.491</td>
<td>-5997.410</td>
<td>-5986.370</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>17</td>
<td>0.057</td>
<td>-2546.523</td>
<td>-2544.360</td>
<td>-2483.502</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>35</td>
<td>0.115</td>
<td>-2297.928</td>
<td>-2288.418</td>
<td>-2168.179</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>43</td>
<td>0.014</td>
<td>-2926.289</td>
<td>-2911.565</td>
<td>-2766.883</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>SINDy</td>
<td>1</td>
<td>10</td>
<td>1.738</td>
<td>-1531.467</td>
<td>-1530.708</td>
<td>-1494.396</td>
<td>0.049</td>
<td>0.053</td>
<td>0.253</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>6</td>
<td>2.480</td>
<td>-1432.442</td>
<td>-1432.156</td>
<td>-1410.199</td>
<td>1.529 ( \times 10^{-23} )</td>
<td>2.094 ( \times 10^{-23} )</td>
<td>1.318 ( \times 10^{-19} )</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>4.619</td>
<td>-1251.263</td>
<td>-1251.182</td>
<td>-1240.142</td>
<td>6.951 ( \times 10^{-63} )</td>
<td>1.055 ( \times 10^{-62} )</td>
<td>1.557 ( \times 10^{-56} )</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>11</td>
<td>1.693</td>
<td>-1537.331</td>
<td>-1536.418</td>
<td>-1496.553</td>
<td>0.914</td>
<td>0.914</td>
<td>0.744</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>12</td>
<td>1.721</td>
<td>-1530.390</td>
<td>-1529.307</td>
<td>-1485.905</td>
<td>0.028</td>
<td>0.026</td>
<td>0.004</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>26</td>
<td>2.531</td>
<td>-1352.359</td>
<td>-1337.636</td>
<td>-1192.954</td>
<td>6.235 ( \times 10^{-41} )</td>
<td>6.255 ( \times 10^{-44} )</td>
<td>8.822 ( \times 10^{-67} )</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>33</td>
<td>2.352</td>
<td>-1312.912</td>
<td>-1308.184</td>
<td>-1220.234</td>
<td>1.694 ( \times 10^{-49} )</td>
<td>2.517 ( \times 10^{-50} )</td>
<td>7.403 ( \times 10^{-61} )</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>10</td>
<td>3.505</td>
<td>-1320.333</td>
<td>-1319.574</td>
<td>-1283.262</td>
<td>6.923 ( \times 10^{-48} )</td>
<td>7.485 ( \times 10^{-48} )</td>
<td>3.595 ( \times 10^{-47} )</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>41</td>
<td>2.527</td>
<td>-1356.817</td>
<td>-1343.529</td>
<td>-1294.826</td>
<td>5.792 ( \times 10^{-40} )</td>
<td>1.186 ( \times 10^{-42} )</td>
<td>3.339 ( \times 10^{-64} )</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>29</td>
<td>2.946</td>
<td>-1444.377</td>
<td>-1437.956</td>
<td>-1336.870</td>
<td>5.972 ( \times 10^{-21} )</td>
<td>3.806 ( \times 10^{-22} )</td>
<td>1.573 ( \times 10^{-35} )</td>
</tr>
</tbody>
</table>

The model selection criteria were able to correctly select the true model in all cases except for the SINDy-SA approach with 10 data points, which resulted in selecting a model with a higher BIC value than the true model. The Pareto curves relating the SSE to the number of model parameters for the SINDy-SA and SINDy methods are shown in Figure SM-5 of the Supplementary Material.

Table 7 shows the model selection results for the 10 and 15 models identified for the SINDy-SA and original SINDy methods, respectively. The corresponding Pareto curves relating the SSE to the number of model parameters are illustrated in Figure SM-5 in the Supplementary Material. Note that both approaches identified a larger number of mathematical models than the other presented applications, but the information criteria were able to correctly select the true model that originated the experimental data for the SINDy-SA method. However, the model selected for the original SINDy approach does not correspond to the true model since it incorporates a source term in the time evolution of the susceptible population \( S(t) \) that is not present in the true model. In both cases, there was great evidence in the choice of the best models. Figure 7 shows a comparison between the experimental data and the numerical solution of the selected best models, whose mathematical description is also shown inside the graphs.

As in the tumor growth model application, the original SINDy approach also fails to solve the system identification problem. Therefore, the results obtained for this application also show that the SA of model parameters in our proposed approach circumvents the difficulty of choosing the threshold and allows the correct identification of the true dynamical system.

3.5 Remarks on Limited and Noise-Contaminated Data Scenarios

It is worth commenting on the robustness of the SINDy-SA approach with respect to scenarios with noisy or sparse data. We carry out studies with the prey-predator (10), the tumor growth (11), and the pendulum motion (12) models to investigate these issues while keeping the other application settings. For simulating scenarios with noisy data, the first two models were perturbed with a multiplicative noise following Log-normal(0, 0.01) and Log-normal(0, 0.1), respectively, while the data of the latter model were contaminated with \( N(0, 0.01) \) additive noise. For investigating scenarios with limited data, we then reduce the number of data from those three applications to \( m = 70, 10, 70 \) evenly distributed time points, respectively. Both the noisy data and such a significant reduction in the number of data points did not prevent the method from identifying the structures of the true models.

With a further increase in the noise level (i.e., increase the standard deviation of the probability distributions) or reduction in the number of simulated data, our approach is not able to solve the dynamical system identification problems. Of note, the original SINDy method is also sensitive to noisy and limited data. In these cases, a possible way to overcome these limitations could be the application of data preprocessing. In particular, we have obtained promising results by using the Gaussian process regression technique [18] in terms of reducing the noise of experimental data and increasing the number of data, both in synthetic and real applications.
Fig. 6: Comparison between the observed data $X$ and the numerical solution $\hat{X}$ of the best pendulum motion models, shown inside the graphs. Although the dynamics are quite similar, the model in (b) includes eight terms (shown in gray) that are not present in the true model.

4 Final Remarks

In this work, we build on the original SINDy method by integrating it with a global SA technique of model parameters. The SINDy method has recently been used to identify nonlinear dynamical system structures from noisy measurement data. However, this method depends on choosing a threshold under which regression coefficients are eliminated in the process of identifying the model structure. The SA technique circumvents the need to define a threshold value by classifying terms according to their importance in relation to the rate of change in time of each state variable (QoI) and eliminating those less influential.

In our proposed SINDy-SA approach, we employed the EE method to analyze the sensitivity of the parameters, and the defined QoI is associated with the formulation of a sparse regression problem. We chose the EE method since it allows ranking the global influence of the model parameters on a QoI and requires a relatively small number of model evaluations compared to variance-based methods. We remark that the proposed methodology is not dependent on the EE method and other global SA techniques can also be used. For solving the sparse regression problem, we used ridge regression, also implemented in the PySINDy module, due to several previous experiments involving using classical regression methods to identify nonlinear dynamical systems. Our SINDy-SA method is carried out through an iterative process, whose convergence is dictated by statistics of the dynamic change of previously computed errors.
Table 7: Model selection results for the SIR application. Subscripts $w_i$ indicate the model selection criterion weights. Models selected as the best ones are indicated in the gray line.

<table>
<thead>
<tr>
<th>Method</th>
<th>Model</th>
<th>$d$</th>
<th>SSE($\mathbf{X}, \hat{\mathbf{X}}$)</th>
<th>AIC</th>
<th>AICc</th>
<th>BIC</th>
<th>AIC $w_i$</th>
<th>AICc $w_i$</th>
<th>BIC $w_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>SINDy-SA</td>
<td>3</td>
<td>0.861</td>
<td>$-2450.414$</td>
<td>$-2450.353$</td>
<td>$-2438.440$</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>29.585</td>
<td>$-1025.674$</td>
<td>$-1025.305$</td>
<td>$-993.742$</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>28.116</td>
<td>$-1046.044$</td>
<td>$-1045.676$</td>
<td>$-1014.113$</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>16</td>
<td>2.799</td>
<td>$-1952.874$</td>
<td>$-1951.453$</td>
<td>$-1889.010$</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>19</td>
<td>2.201</td>
<td>$-2041.051$</td>
<td>$-2041.051$</td>
<td>$-1967.214$</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>29</td>
<td>2.318</td>
<td>$-2002.284$</td>
<td>$-1997.582$</td>
<td>$-1886.532$</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>35</td>
<td>2457.726</td>
<td>4480.347</td>
<td>4487.270</td>
<td>4620.048</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>41</td>
<td>2310.975</td>
<td>4467.712</td>
<td>4477.332</td>
<td>4631.362</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>37</td>
<td>2326.156</td>
<td>4462.341</td>
<td>4470.109</td>
<td>4610.025</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>40</td>
<td>2326.194</td>
<td>4468.346</td>
<td>4477.482</td>
<td>4628.004</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>SINDy</td>
<td>1</td>
<td>8</td>
<td>25.032</td>
<td>$-1092.525$</td>
<td>$-1092.156$</td>
<td>$-1060.593$</td>
<td>$3.799 \times 10^{-279}$</td>
<td>$3.324 \times 10^{-279}$</td>
<td>$1.296 \times 10^{-282}$</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>6.667</td>
<td>$-1628.340$</td>
<td>$-1628.188$</td>
<td>$-1608.383$</td>
<td>$8.523 \times 10^{-163}$</td>
<td>$8.308 \times 10^{-163}$</td>
<td>$1.158 \times 10^{-163}$</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>3300.455</td>
<td>5450.261</td>
<td>5459.413</td>
<td>5479.218</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>2651.547</td>
<td>4446.709</td>
<td>4446.770</td>
<td>4458.684</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>25.405</td>
<td>$-1090.612$</td>
<td>$-1090.398$</td>
<td>$-1066.663$</td>
<td>$1.460 \times 10^{-279}$</td>
<td>$1.380 \times 10^{-279}$</td>
<td>$2.697 \times 10^{-281}$</td>
<td></td>
</tr>
<tr>
<td>SINEy</td>
<td>6</td>
<td>4</td>
<td>1.035</td>
<td>$-2374.697$</td>
<td>$-2374.796$</td>
<td>$-2358.732$</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>2960.483</td>
<td>5934.021</td>
<td>5934.031</td>
<td>5398.013</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>7.000</td>
<td>$-1610.199$</td>
<td>$-1610.098$</td>
<td>$-1594.233$</td>
<td>$9.800 \times 10^{-167}$</td>
<td>$9.800 \times 10^{-167}$</td>
<td>$9.800 \times 10^{-167}$</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>26.101</td>
<td>$-1081.792$</td>
<td>$-1081.639$</td>
<td>$-1061.834$</td>
<td>$1.774 \times 10^{-291}$</td>
<td>$1.730 \times 10^{-291}$</td>
<td>$2.412 \times 10^{-292}$</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>7845.297</td>
<td>4884.772</td>
<td>4904.577</td>
<td>4904.577</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>3</td>
<td>37923.834</td>
<td>5655.790</td>
<td>5654.790</td>
<td>5654.790</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>5</td>
<td>51561.874</td>
<td>5637.765</td>
<td>5637.918</td>
<td>5657.723</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>2</td>
<td>37835.247</td>
<td>11034.198</td>
<td>11034.229</td>
<td>11042.181</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>0</td>
<td>28625.888</td>
<td>5392.374</td>
<td>5392.374</td>
<td>5392.374</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>2</td>
<td>24466.197</td>
<td>5331.569</td>
<td>5333.600</td>
<td>5341.552</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
</tbody>
</table>

Both SINDy-SA and SINDy approaches were executed in a variety of applications, in which we generated simulated data from a true model in order to analyze the ability of the methods to correctly discover the structure of the mathematical model, as well as the parameter values. The applications presented include models with different behaviors encompassing a prey-predator model, a logistic model calibrated from tumor growth data, a pendulum motion model, and a SIR compartmental model. For each example, we elaborated different experimental settings yielding a set of resulting models, whose best model was selected using information criteria.

In the prey-predator and pendulum motion applications, the original SINDy method correctly discovered the structure of the dynamical system, although not selected by the information criteria in the pendulum application. However, the parameter values of the discovered models were approximated. Thus, the results obtained for these applications highlighted the importance of a model recalibration step to improve the accuracy of the parameter estimates. In the logistic growth and compartmental applications, the SINDy method wrongly incorporated additional terms to the true mathematical model, showing a limitation of using a threshold to eliminate candidate functions. On the other hand, our combined approach was able to correctly identify the true model that generated the available data in all applications, emphasizing a relevant advantage of performing a SA of model parameters to solve the system identification problem. These facts demonstrate the potential of our SINDy-SA method to be used in real applications, in which the true model is not known a priori, in the search for interpretable and predictive mathematical models.

The time and state variables to perform a global sensitivity analysis can play a crucial role in model identification. For problems with multiple dependent outputs, Saltelli et al. [30] suggest composing multiple outputs into a single scalar and performing a sensitivity analysis. However, in some situations, it is more appropriate to quantify the sensitivity of the parameters at each output [19]. Although this analysis was not carried out in the present work, applying multi-output global sensitivity analysis methods in our framework is the target of future investigations.

Overall, our SINDy-SA approach can leverage new developments together with other recent SINDy works to deal with, for example, implicit dynamical systems [16], partial differential equations [29] and control [3], and with data preprocessing techniques [18] to provide robustness on system identification, even in scenarios with limited and noisy data.

**Declarations**

**Funding**

The authors would like to thank Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES).
Fig. 7: Comparison between the observed data $X$ and the numerical solution $\hat{X}$ of the selected best SIR models, shown inside the graphs. Although the dynamics are quite similar, the model in (b) includes one term (shown in gray) that is not present in the true model.

Gustavo T. Naozuka is supported by a PhD scholarship from Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro (FAPERJ), grant number 200.325/2020. Heber L. Rocha is supported by a postdoctoral fellowship from U01-CA232137 (NCI), 1720625 (NSF), 1818187 (NSF), the Breast Cancer Research Foundation, and the Jayne Koskinas Ted Giovanis Foundation for Health and Policy.

Conflict of Interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be constructed as a potential conflict of interest. The authors have no affiliation with any organization with a direct or indirect financial interest in the subject matter discussed in the manuscript.

Availability of Data and Material

The data used in the current study are generated by running the source code publicly available at https://github.com/tmglncc/SINDy-SA.

Code Availability

The source code used to generate the results during the current study is publicly available at https://github.com/tmglncc/SINDy-SA.
References

fr/hal-03130833


Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- SupplementaryMaterial.pdf