Structure is information: structural identifiability mappings for machine learning with partially observed dynamical systems

Janis Norden, Elisa Oostwal, Michael Chappell, Peter Tiňo and Kerstin Bunte

***Abstract*—The successful application of modern machine learning for time series classification is often hampered by limitations in quality and quantity of available training data. To overcome these limitations, domain knowledge can be leveraged in the form of parameterised mechanistic dynamical models, whereby time series observations may be represented as instances of a predefined class of dynamical systems. Provided the dynamical models are interpretable in terms of domain-specific variables and their dynamic interaction, the learning process becomes interpretable as well and enables the modeller to handle sparsely and irregularly sampled data naturally. However, the internal processes of a dynamical model are often only partially observed. This can lead to ambiguity regarding which particular model realization best explains a given time series observation. This problem is well-known in the literature, and a dynamical model with this issue is referred to as structurally unidentifiable. Training a classifier that ignores knowledge about a structurally unidentifiable dynamical model can negatively influence classification performance. To address this issue, we employ structural identifiability analysis to explicitly relate parameter configurations that are associated with identical system outputs. Using the derived relations in classifier training, we demonstrate that this method significantly improves the classifier's ability to generalize to unseen data on a number of example models from the biomedical domain. This effect is especially pronounced when the number of training instances is limited. Our results demonstrate the importance of structural identifiability, a topic that has received relatively little attention from the machine learning community.**

I. Introduction

THE problem of time series classification concerns the assignment of an observed time series to one of a predefined set of classes. As time series data naturally arise in a wide variety of scientific disciplines, including medicine, engineering and the social sciences, the associated theory of classification has been applied with great success to a multitude of problems such as health monitoring [1], maintenance of civil infrastructure [2] and emotion recognition from speech [3]. While research has been directed at this problem over many years, reliable and time-efficient classification of time series remains a challenging problem to date [4]–[6].

In many disciplines, the classification task is accompanied by domain-specific knowledge in the form of a mechanistic model which describes the data-generating mechanism. This kind of model is typically derived from the application of a physical, chemical, biological or sociological law, and often takes the form of a parametrised dynamical system model. If such a model is available, then its incorporation into the classification task is beneficial in two ways: Firstly, interpreting the data in the context of the mechanistic model makes it possible to deal with sparsely and irregularly sampled time series data in a natural way. Secondly, any learned classification rule becomes interpretable as it directly relates to the given mechanistic model [7]. The second point is of particular importance for the modelling of high-risk applications common to biomedical and engineering domains, where the interpretability of the application of machine learning is of critical importance. When safety, correctness and trustworthiness need to be guaranteed, model-based approaches are often the only viable option.

Model- and data-driven hybrid techniques, often referred to as scientific or physics-informed machine learning, have emerged to leverage the scientific knowledge in order to analyse, simulate, and predict the behaviour of complex physical, biological, or engineering systems, with a recent review provided in [8]. This typically includes the use of physical laws, constraints, or symmetries, which are embedded into machine learning models to improve data efficiency, interpretability, and generalization, especially in regimes with limited or noisy data [9]–[11]. Beyond prioritizing accuracy, biological plausibility and realism are increasingly becoming central objectives in time series analysis [12]. Assuming that a mechanistic model in the form of a parametrised dynamical system model is indeed available, then it is not automatically guaranteed that model-based classification works well. The degree to which variables in the dynamical model can be observed is often restricted by practical and ethical considerations. This limited observability may lead to ill-posed parameter estimation problems when trying to infer model parameters from given time series data, which is also identified as core failure mode of physics-informed neural networks applied to dynamical systems [13], [14]. Often, these degeneracies are unrecognized, or masked by optimization choices and the explicit integration of symmetry awareness, or structure-informed constraints, are still an open and active research area [15].

The problem of determining whether a given model allows for unique inference of model parameters has been studied extensively in the literature over the past 50 years and is known as Structural Identifiability (SI) [16]. If a given model is not structurally identifiable, then multiple parameter configurations will produce identical input-output behaviour of the model. It follows that parameters cannot be meaningfully estimated, regardless of the amount and quality of the available data. SI is closely related to the well-known control-theoretic

J. Norden is with the Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence, University of Groningen, The Netherlands

concept of observability. A dynamical system is called observable, if it is possible to determine its internal state from measurements of the output [17]. When system parameters are viewed as state variables with zero dynamics, SI analysis may be regarded as a generalized observability problem [18]. In the following, we adopt the notion of "partially observed" as introduced in [7], denoting systems for which only a subset of its state variables (or combinations thereof) can be measured. Note the principal difference between the notions of "partially observed" and "partially observable" systems, the latter being concerned with the possibility of estimating the internal state from measured system output. SI is distinguished from Practical Identifiability (PI), which addresses ambiguities in parameter estimation due to noise and unfavourable sampling times of observations. Structural and practical identifiability are connected to epistemic (systematic) and aleatoric (stochastic) uncertainty [19]. The discussion about the importance of these uncertainty categories reignited [20] and distinct handling has gained traction in the machine learning community [21]. Therefore, SI analysis constitutes an important component for the understanding and reduction of epistemic uncertainty for dynamical systems.

A recent review [22] finds that the discipline of SI analysis can be separated into three main branches: the Output Equality Approach [23]–[25], the Local State Isomorphism Approach [26]–[28], and the Differential Algebra Approach [29]–[31]. Each technique has its respective strengths and limitations. In particular, the Output Equality Approach is not generally applicable to non-linear models, but offers an intuitive way for analysing linear models.

In cases where a model is unidentifiable, a range of strategies is commonly used to make parameter inference from data feasible. A straightforward option is to fix selected parameters so that the remaining set becomes identifiable. This approach is simple and preserves the interpretability within the domain-specific context. Considerable disadvantages include the necessity for profound mechanistic insight into the model used and a reduced potential for interpretation of the model predictions [32]. Another option is to approximate the behaviour of the unidentifiable model with an alternative identifiable model, which is typically nontrivial. Finally, one may attempt to reparametrise an unidentifiable model such that all of the new parameters in the resulting model become identifiable. Reparametrisation of a given model often requires a time-consuming manual effort in which practitioners enter a cycle of model construction, quantitative simulations and experimental validation of model predictions. Recent advances in automatic reparametrisation for dynamical models show great potential [33], [34]. Notably, the *AutoRepar* extension [35] for the STRIKE GOLDD SI analysis toolbox [18] for MATLAB is capable of semi-automatic reparametrisation for ordinary differential equation models involving rational expressions.

`AutoRepar` employs a notion of identifiability called Full Input-State-Parameter Observability (FISPO). It goes beyond establishing parameter identifiability and requires that all states of some auxiliary model are observable. However, requiring a model to be FISPO is a stronger condition than identifiability alone, and thus `AutoRepar` is not generally applicable. It works by detecting and eliminating Lie symmetries that can cause unidentifiability. However, this approach has two major limitations: (i) no general method exists to determine the type and number of symmetries in a given model, and (ii) there is no upper bound on the number of Lie derivative terms required to construct the infinitesimal transformation needed for a suitable reparametrisation [36]. In summary, even with semi-automatic reparametrisation tools such as `AutoRepar`, deriving a reparameterised model that is both identifiable and retains domain-specific interpretability remains challenging.

As reparameterising a given dynamical model is often highly involved, whereas structural identifiability analysis can frequently be carried out more readily, we propose a model-based framework for time series classification that accounts for the unidentifiability of the underlying dynamical model, referred to as "Structural Identifiability Mapping" (SIM). We employ a model-based time series classification where each individual time series is represented through a Maximum A Posteriori estimate (MAP) of the given dynamical model. We consider Ordinary Differential Equation (ODE) models in which one or more parameters are unidentifiable. Instead of representing individual time series as parameter vectors in the original parameter space, we consider a representation in the space of structurally identifiable parameter combinations. Any conventional classification framework acting on vectorial data may subsequently be used to train a classifier in this space.

The contribution of this work is threefold. Firstly, we propose a novel framework (SIM) for time series classification that represents time series data as identifiable parameter combinations of an otherwise unidentifiable dynamical system. Second, we demonstrate the effectiveness of this framework on several relevant dynamical system models commonly encountered in computational biology. In particular, we demonstrate that explicitly accounting for model unidentifiability enables accurate classification even with limited training data. Finally, we emphasize the importance of SI analysis whenever machine learning is applied in conjunction with parametrised dynamical system models, an aspect that has received limited attention despite its critical role in ensuring successful application.

This paper is organized as follows: section II reviews a model-based framework for time series classification and introduces our method of Structural Identifiability Mapping (SIM). section III presents three biologically relevant example models used as test beds for SIM and outlines the experiments used to evaluate its performance. section IV reports the experimental results. Finally, section V and section VI provide a discussion of the results and their implications.

II. Methods

In this section, we present a model-based approach for time series classification based on the incorporation of a given dynamical model in the form of a set of parametrised Ordinary Differential Equations (ODEs). To do so, we adopt a formalism in which individual time series observations are represented as Maximum A Posteriori (MAP) estimates. In addition, we present the details of the proposed strategy, namely a **S**tructural-**I**dentifiability **M**apping (SIM). The application

of a SIM is possible whenever the underlying dynamical model is structurally unidentifiable, then structural identifiability analysis can be carried out and explicit expressions for identifiable parameter combinations can be determined. This notably includes the class of non-linear ODE models with rational expressions of the states, inputs, and parameters, for which software tools such as *SIAN* [33], *COMBOS* [37] and *Structural-Identifiability* [34] may be used to automatically determine identifiable model parameter combinations [38].

A. Model-based representation for time series data

In the following, we review the basic notions of Bayesian parameter estimation for dynamical models and adapt them for the purposes of time series classification. Formulations similar to the one given in this work can be found in [7], [39], [40].

Let $\{(\mathcal{Y}^k, c^k)\}, k = 1, \ldots, N$, denote a set of N labelled examples of, potentially multivariate, time series data. Here $\mathcal{Y}^k = \{\mathbf{t}^k, \mathbf{Y}^k\}$ consists of a collection of time points $\mathbf{t}^k = \{t_i^k : i = 1, \ldots, L^k\}$ together with a collection of the corresponding observations $\mathbf{Y}^k = \{\mathbf{y}_i^k : i = 1, \ldots, L^k\}$ for the time series k. Furthermore, c^k is the associated class label. This formulation allows for different time series $\mathcal{Y}^k$ to be of different lengths, as indicated by L^k, and be evaluated at different times, as indicated by $\mathbf{t}^k$. However, it is assumed that all observations have the same dimension, i.e., $\mathbf{y}_i^k \in \mathbb{R}^r$. *The task considered is the prediction of a class label c, given a new time series $\mathcal{Y}$ of length L.* The key idea of this framework is to regard each time series as an instance of a dynamical model from a given model class. Time series are considered as partial observations of an underlying dynamical model characterized by a set of Ordinary Differential Equations (ODEs)

$$\frac{d\mathbf{x}_t}{dt} = f(\mathbf{x}_t; \boldsymbol{\psi}), \tag{1}$$

with $\mathbf{x}_t \in \mathbb{R}^d$ denoting the state vector at time t. The defining mapping f is parametrized by a vector $\boldsymbol{\psi} = (\boldsymbol{\theta}, \mathbf{x}_0)$, where $\boldsymbol{\theta} \in \mathbb{R}^n$ is a vector of model parameters and the initial state $\mathbf{x}_0$, which may or may not be known. Observations from the underlying ODE are obtained via the measurement function

$$\mathbf{y}_i = \mathbf{h}(\mathbf{x}_{t_i}) + \boldsymbol{\epsilon}_{t_i}, \tag{2}$$

where $\boldsymbol{\epsilon}_{t_i}$ is the observational noise at time t_i.

For simplicity, it is assumed that the initial condition vector $\mathbf{x}_0$ is known and that the observational noise is distributed as $\boldsymbol{\epsilon}_{t_i} \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$, i.e. Gaussian with zero mean and covariance matrix $\mathbf{R}$. Should $\mathbf{x}_0$ be unknown, it may be treated as an additional unknown system parameter. If $\mathbf{R}$ is unknown, it may be estimated from the variation of the observed measurements. The parameter configuration that is most likely to have produced an observation $\mathcal{Y}$, given a prior $p(\boldsymbol{\theta})$ over the parameters, is the *Maximum A Posterior* (MAP) estimate $\boldsymbol{\theta}_{\text{MAP}}$, which is the (global) maximum of the posterior distribution

$$p(\boldsymbol{\theta} \mid \mathcal{Y}, \mathbf{R}) = p(\boldsymbol{\theta} \mid \mathbf{Y}, \mathbf{t}, \mathbf{R}) \propto p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathbf{t}, \mathbf{R})\ p(\boldsymbol{\theta}). \tag{3}$$

Under the assumptions made in Eq.(2), the likelihood function takes on the form

$$p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathbf{t}, \mathbf{R}) = \prod_{i=1}^{L} \mathcal{N}(\mathbf{y}_i \mid \mathbf{x}_t(\boldsymbol{\theta}), t_i, \mathbf{R}). \tag{4}$$

Finally, for the purposes of this work, we assume that the prior distribution is of the "bounding box" form

$$p(\boldsymbol{\theta}) = \begin{cases} \frac{1}{V(R)} & \text{if } \boldsymbol{\theta} \in R, \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

where $R = [\theta_1^{\min}, \theta_1^{\max}] \times \ldots \times [\theta_n^{\min}, \theta_n^{\max}]$ is the hyper-rectangle enclosed by the individual parameter bounds $\theta_i^{\min}, \theta_i^{\max}$ and $V(R)$ is the volume of R. The set R will be referred to as Region of Interest (ROI). This prior information essentially restricts the considered region of the parameter space to R but does not provide any additional information, i.e., is uniform over the region R. Interval priors are quite common in biological models, since often only "physiologically realistic" parameter ranges are known without further probabilistic structure. In order to find the $\boldsymbol{\theta}_{\text{MAP}}$ associated with a given time series observation, Eq. (3) is maximized w.r.t. $\boldsymbol{\theta}$, which is equivalent to maximizing Eq. (4) subject to $\boldsymbol{\theta} \in R$.

B. Structural-Identifiability Mapping (SIM)

Suppose that the dynamical model given in Eq. (1) is unidentifiable and that, by means of Structural Identifiability (SI) analysis, it is possible to find a set of identifiable parameter combinations $\boldsymbol{\Phi}$ explicitly characterized by $\boldsymbol{\Phi} = g(\boldsymbol{\theta})$, with $g : \mathbb{R}^n \to \mathbb{R}^m$. Here, the number of identifiable parameter combinations m is always less than the number of original system parameters n, i.e. $m < n$. Consider an equivalence relation on the space of our mechanistic models that identifies models that are behaviourally indistinguishable. The equivalence classes of models (parameters) $\mathcal{M}_{\boldsymbol{\Phi}}$ can be then defined as follows:

$$\mathcal{M}_{\boldsymbol{\Phi}} = \{\boldsymbol{\theta} \in \mathbb{R}^n \mid \boldsymbol{\Phi} = g(\boldsymbol{\theta})\}. \tag{6}$$

By definition of g, any two parameters $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2 \in \mathcal{M}_{\boldsymbol{\Phi}}$ will lead to identical system trajectories of the system in Eq. (1), given identical initial conditions. We can operate in the factor set. Indeed, any level-set of the posterior in Eq. (3) can be written as a union of sets $\mathcal{M}_{\boldsymbol{\Phi}}$ and maximization of the posterior means to identify the set of equivalence classes associated with the maximal posterior value. As far as the classification task is concerned, there is no need to resolve the available information beyond the level of equivalence classes. Figure 1 provides some visual intuition on the matter. We propose to utilize Structural Identifiability Mapping (SIM) given by g for time series classification as follows:

1) Find the model-based representation for each time series by means of a MAP estimate, i.e.

$$\mathcal{Y}^k \mapsto \boldsymbol{\theta}_{\text{MAP}}^k, \tag{7}$$

$$\text{with} \quad \boldsymbol{\theta}_{\text{MAP}}^k = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta} \mid \mathcal{Y}^k, \mathbf{R}), \tag{8}$$

and posterior as in Eq. (3).

2) Translate each MAP via g to obtain a representation in the space of identifiable parameter combinations

$$\boldsymbol{\theta}_{\text{MAP}}^k \mapsto \boldsymbol{\Phi}^k := g(\boldsymbol{\theta}_{\text{MAP}}^k). \tag{9}$$

3) Train a vectorial classifier of choice on the transformed data $\{\boldsymbol{\Phi}^k\}_{k=1}^N$.

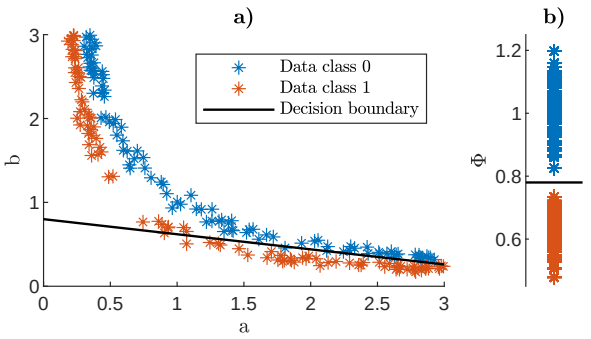

Fig. 1. Geometric intuition behind the mechanism of SIM with data from the toy model. Panel **a)** depicts a binary classification problem which illustrates how training data can be oriented along manifolds of the form $\Phi = g(a, b) = ab$. Panel **b)** shows the representation of the same data after applying SIM. The decision boundary between the two classes becomes simpler and, in this special case, the data even become linearly separable in Φ-space.

How is the application of SIM different from reparametrising a given dynamical model in order to make it structurally identifiable? The answer is that SIM can *always* be used when structurally identifiable combinations of parameters can be computed. However, the reparametrisation of a given model in terms of such a set of structurally identifiable combinations is *not always* possible. In this sense, SIM focuses on the ML task at hand rather than the creation of an all-new dynamical model with more favourable identifiability properties.

From a classification point of view, SIM can be thought of as having a regularizing influence on the learned decision boundary in the $\boldsymbol{\theta}$-space. If we were to train the classifier in the space of $\boldsymbol{\theta}$, the decision boundary learned from the data could be such that two values $\boldsymbol{\theta}_1 \neq \boldsymbol{\theta}_2$ with $g(\boldsymbol{\theta}_1) = g(\boldsymbol{\theta}_2)$ become associated with different classes. This results in undesired behaviour, since SI analysis tells us that both values of $\boldsymbol{\theta}$ will yield identical observable output for our dynamical model and should therefore be associated with the same class. On the other hand, training the classifier using SIM, the learned decision boundary in $\boldsymbol{\theta}$-space becomes the union of pre-images $g^{-1}(\boldsymbol{\Phi})$. This guarantees that any two models $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2$ with $g(\boldsymbol{\theta}_1) = g(\boldsymbol{\theta}_2)$ are always associated with the same class.

III. Experiments

Investigation of SIM is carried out through three experiments. Experiment 1 compares learning with a partially observed dynamical system when SIM, using a fully observed counterpart of the same dynamical system as a baseline. Robustness of SIM with respect to observational noise is studied in experiment 2. Experiment 3 addresses the robustness of SIM with respect to sparsity and irregularity in the time series observations. These experiments are performed on synthetic data generated by four example systems of increasing complexity, as introduced in subsection III-A, detailed in subsection III-B, and summarized in Table I. MATLAB implementations are publicly available on Github[1].

[1]https://github.com/janis-norden/Structural_Identifiability_Mapping

TABLE I
Summary of synthetic example models and experiments. FO≙fully observed and PO≙partially observed.

System	Characteristics	Exp.	Purpose
toy	simplest illustrative model		
CCM2	simplest model appearing in applications	1:	compare FO vs. PO vs. PO + SIM approaches
CCM4	larger variant of CCM2		
CML	cannot be reparameterised by scaling transformation	2:	assess impact of observation noise on SIM
BR	nonlinear model and unknown initial conditions	3:	assess impact of measurement sparsity and irregularity

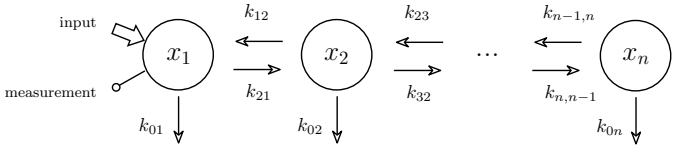

Fig. 2. Catenary n-compartmental model.

A. Example Models

1) Toy Model: is a simple artificial model which allows for intuitive visualization of the SIM due to its 2-dimensional parameter space (Figure 1). The model equations are given by

$$\begin{aligned} \dot{x}(t) &= -abx(t), \\ y(t) &= x(t), \end{aligned} \tag{10}$$

where $x \in \mathbb{R}$ is the state variable with $x(0) = 1$ known, $t \in [0, 1]$, and $a, b \in \mathbb{R}^+$ are the system parameters. The parameters are further restricted to lie within the region of interest (ROI) $R = [0.1, 3] \times [0.1, 3]$. From Eq. (10) it can be seen that any parameter configuration a and b such that $\Phi = ab$ is constant, will produce identical system output for a given value of Φ.

2) Catenary compartmental Model (CCM): Compartmental models are commonly used in modelling pharmacokinetic interactions [41]–[43]. The n-compartment catenary model (CMMn), is a linear model of n compartments which are connected to one another in a bi-directional chain. Only the first compartment is assumed to have an input, whereas all compartments are assumed to have leakage (see Figure 2). Substance x_i is converted to x_{i+1} and vice versa. The model has a total of $3n - 2$ parameters comprising $2(n - 1)$ conversion rates and n leakages. The coefficients $k_{i,i-1} \geq 0$ and $k_{i-1,i} \geq 0$ describe the conversion rates between x_i and x_{i-1}, while the coefficients $k_{0i} \geq 0$ govern the leakage. The concentration of interacting substances $x_1, \ldots, x_n$ is described by the set of linear ODEs:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= K\mathbf{x}(t) + \mathbf{b}u(t) \\ y(t) &= x_1(t), \end{aligned} \tag{11}$$

where $\mathbf{b} = [1, 0, \ldots, 0]^\top$, $\mathbf{x} = [x_1, \ldots, x_n]^\top$ with $\mathbf{x}(0) = \mathbf{x}_0$, and y is the system output. The matrix K is then given by

$$K = \begin{bmatrix} k_{11} & k_{12} & 0 & 0 & \ldots & 0 \\ k_{21} & k_{22} & k_{23} & 0 & \ldots & 0 \\ 0 & k_{32} & k_{33} & k_{34} & \ldots & 0 \\ 0 & 0 & k_{43} & k_{44} & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & k_{n,n-1} & k_{n,n} \end{bmatrix}, \text{ with} \quad (12)$$

$$k_{ii} = \begin{cases} -k_{01} - k_{21}, & \text{for} \quad i = 1, \\ -k_{0i} - k_{i+1,i} - k_{i-1,i} & \text{for} \quad i = 2, 3, \ldots, n-1, \\ -k_{0n} - k_{n-1,n} & \text{for} \quad i = n. \end{cases} \quad (13)$$

Employing the Laplace transform Output Equality approach, Chen et al. [44] demonstrated that $2n-1$ structurally identifiable parameter combinations can be found for CCM, namely:

$$\Phi_i^1 = k_{ii}, \qquad \text{for} \quad i = 1, 2, \ldots, n, \quad (14)$$

$$\Phi_j^2 = k_{j,j-1} k_{j-1,j}, \qquad \text{for} \quad j = 2, 3, \ldots, n. \quad (15)$$

Using AutoRepar, we were able to reparametrise the CCM2 model to a FISPO model. The same is not true for the CCM4 model (see Appendix A for details).

We consider the work by Bunte et al. [45] for experimentation, in which the data from a clinical study concerning the interaction between the metabolites prednisone and prednisolone is analysed. The authors employed a 3-compartment model and used a probabilistic mixture of such models to analyse the data of 12 patients and found that the patients could be stratified into 4 groups. Their 3-compartment model is equivalent to a CCM with 2 compartments with non-zero input and is therefore suitable for the application of a SIM. The input in this case is $u(t) = S_0 k_{\text{abs}} e^{-k_{\text{abs}} t}$, where S_0 is a fixed amount of prednisone formulation that is ingested and absorbed with rate k_{abs} into the bloodstream. The time interval of interest is $t \in [0, 240]$ seconds and the ROI is $R = [0, 0.1]^4$.

To set up a suitable binary classification problem, we use the parametrisation of one of the clusters (C4) found in [45] to represent one of the classes. The other class is characterized by the same parametrisation, where the conversion rates k_{12} and k_{21} are 20% deficient (see Table II). This model will be referred to as CCM2 and is of particular interest, since it represents a minimal realistic compartmental model for which structurally unidentifiable parameters occur.

We further consider a 4-compartment variant of this model by adding two additional compartments in accordance with the model schematic in Figure 2. For the first class, the excretion and conversion are the same as those used for the CCM2. The second class is characterized by the same parametrisation but now six out of seven conversion rates are set to be 50% deficient. The studied time interval is the same as for CCM2 and the ROI is $R = [0, 0.1]^{10}$.

3) Compartmental Model with a Loop (CML): To further demonstrate SIM with models which cannot be meaningfully reparametrised in a straightforward manner, the Compartmental Model with a Loop (CML) is considered (see Figure 3). Similar to the CCM, the CML is a linear compartment model

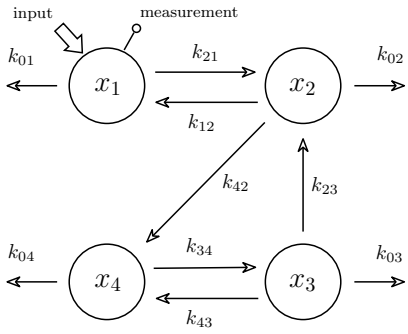

Fig. 3. Compartment Model with a Loop (CML).

and dynamics Eq. (11) apply with coefficient matrix

$$K = \begin{bmatrix} k_{11} & k_{12} & 0 & 0 \\ k_{21} & k_{22} & k_{23} & 0 \\ 0 & 0 & k_{33} & k_{34} \\ 0 & k_{42} & k_{43} & k_{44} \end{bmatrix}, \quad (16)$$

$$\begin{aligned} \text{where} \quad k_{11} &= -(k_{01} + k_{21}), \\ k_{22} &= -(k_{02} + k_{12} + k_{42}), \\ k_{33} &= -(k_{03} + k_{23} + k_{43}), \\ k_{44} &= -(k_{04} + k_{34}). \end{aligned} \quad (17)$$

and the ROI is given as $R = [0, 0.1]^{10}$.

Employing the Laplace transform approach, it can be shown that the system has 7 structurally identifiable parameter combinations $\Phi_1, \ldots, \Phi_7$. The relations are

$$\begin{aligned} \Phi_1 &= k_{12} k_{21}, & \Phi_2 &= k_{34} k_{43}, \\ \Phi_3 &= k_{01} + k_{21}, & \Phi_6 &= k_{04} + k_{34} \\ \Phi_4 &= k_{02} + k_{12} + k_{42}, & \Phi_5 &= k_{03} + k_{23} + k_{43}, \\ \Phi_7 &= k_{23} k_{42} k_{34}. \end{aligned} \quad (18)$$

Meshkat & Sullivant [46] demonstrate that for this system (Example 6.3 in their work) no scaling transformations exist which make the resulting reparametrised system identifiable. We tried AutoRepar with an univariate Ansatz polynomial of degree 2 but could not find any transformations that would make the reparametrised model FISPO. Yet, using the same Ansatz polynomial, we were able to find the relations given in Eq. (18) by only looking at parameter identifiability. Since the CML could not easily be reparametrised, the model is particularly interesting as a test case for SIM.

4) Batch reactor (BR): A classical model defined to study microbial growth in a batch reactor which incorporates a Michaelis-Menten type nonlinearity is the following:

$$\begin{aligned} \dot{x}(t) &= \frac{\mu_m s(t) x(t)}{K_s + s(t)} - K_d x(t), \\ \dot{s}(t) &= -\frac{\mu_m s(t) x(t)}{Y(K_s + s(t))}, \\ y(t) &= x(t), \end{aligned} \quad (19)$$

where x is the concentration of microorganisms, s the concentration of growth-limiting substrate, μ_m the maximum reaction velocity, K_s the Michaelis-Menten constant, Y the

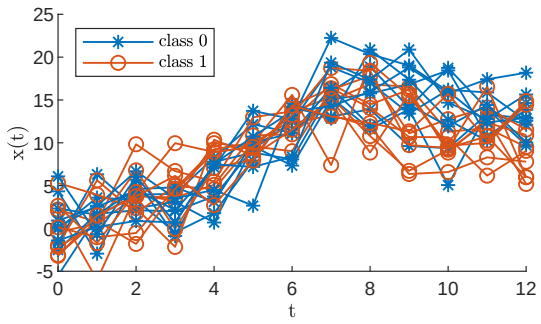

Fig. 4. Binary classification task for time series from the batch reactor model. Displayed are 10 time series per class. Observational noise simulated is normally distributed with standard deviation $\sigma = 3$.

yield coefficient, and K_d the decay rate coefficient (see e.g. [47]). For the present work, the time interval of interest is $t \in [0, 12]$ hours and the ROI is:

$$R = [0, 10] \times [0, 50] \times [0, 1] \times [0, 5] \times [0, 1] \times [0, 1], \quad (20)$$

where the intervals refer to the allowed ranges of b_1, b_2, μ_m, K_s, Y and K_d, respectively.

It is assumed that microorganisms x and substrate s are prepared in mixtures for which the concentration can be controlled. When the mixtures are put together in the batch reactor, it is assumed that the reaction is very fast, so that the model may be regarded as having impulsive inputs $b_1\delta(t)$ for x and $b_2\delta(t)$ for s. Equivalently, system Eq. (19) may be considered with initial conditions $x(0) = b_1$ and $s(0) = b_2$. As demonstrated in [48], if both x and s are observed at time $t = 0$, then the model is globally structurally identifiable. However, in [49], [50] it was demonstrated that when only x is observed, the model becomes structurally unidentifiable. In this case, the following combinations of parameters have been found to be structurally identifiable

$$\begin{aligned} \Phi_1 = b_1, \quad \Phi_2 = \mu_m, \quad \Phi_3 = K_d, \\ \Phi_4 = b_2 Y, \quad \Phi_5 = \frac{b_2}{K_s}. \end{aligned} \quad (21)$$

Realistic configurations of model parameters have been taken from [48] (cf. Figure 1 in their work). As a classification task, we consider a scenario in which two reactions are compared which primarily differ in their yield coefficient Y. Class 0 is characterized by a distribution of yield coefficients centred around $Y = 0.6$ while class 1 is associated with a distribution that centres around a 20% diminished yield coefficient, i.e., $Y = 0.48$. Figure 4 illustrates the classification task in the signal space of time series.

B. Experimental setup

In order to test the effectiveness of SIM approach, a binary classification is implemented based on the Support Vector Machine (SVM) framework. For each example system, synthetic time series data corresponding to a binary classification task are created and the SVM classifier is trained using the discussed model-based framework. The performance of the resulting classifier is assessed by its generalization error. Additionally, the number of support vectors is considered as an indication of the classifier-complexity needed to distinguish the two classes. Since the example systems differ in the dimensionality of their parameter spaces, their training and test sets contain differing numbers of training examples N_{train} and test examples N_{test}. For each system, N_{train} and N_{test} are chosen to be sufficiently large as not to be the limiting factor in the assessment of classification performance.

1) Experiment 1: This experiment compares classification performance for three situations: training with the fully observed (FO) dynamical model, training with the partially observed (PO) dynamical model, and training with the partially observed dynamical model together with a SIM (PO + SIM). The synthetic data $\mathbb{D} = \{(\mathcal{Y}^k, c^k) : \quad k = 1, \ldots, N\}$ used for this experiment are generated as follows.

The ground truth class-conditional distributions associated with classes $c = 0$ and $c = 1$ are chosen as multivariate normal distributions with known means and covariance matrices:

$$p(\boldsymbol{\theta} \mid c_i) = \mathcal{N}(\boldsymbol{\theta}, \boldsymbol{\mu}_i, \Sigma_i), \quad i \in \{0, 1\}. \quad (22)$$

The values of $\boldsymbol{\mu}_i$ and Σ_i used for experimentation are specific to the dynamical model under consideration (see Table II).

An equal number of example pairs $(\mathcal{Y}^k, c^k)$ is generated for each class by first drawing $\boldsymbol{\theta}$ from the associated class-conditional distribution $p(\boldsymbol{\theta} \mid c^k)$ and subsequently integrating the dynamical system Eq. (1) with the drawn $\boldsymbol{\theta}$ on the time interval $t \in [0, t_{\text{end}}]$. To obtain $\mathcal{Y}$, time points are sampled from $[0, t_{\text{end}}]$ and the trajectory of the dynamical system is evaluated at these time points. This leads to a classification problem with balanced classes. For the fully observed dynamical model, the system output mapping is assumed to be the identity, i.e. $h(\mathbf{x}) = \mathbf{x}$, and data for each state variable are generated. For the partially observed model, the system output mapping is the projection onto the first state variable $h(\mathbf{x}) = x_1$.

For experiment 1, a regular time grid on which the data are generated is chosen to densely cover the time interval of interest: t_{dense}. The collection of observations $\mathbf{Y}^k$ is then obtained by evaluating the resulting trajectory $\mathbf{x}(t; \boldsymbol{\theta})$ on the time grid and adding observational noise, which is assumed to be Gaussian, i.e.,

$$\mathbf{y}_i^k = \mathbf{x}(t_i^k) + \boldsymbol{\epsilon}, \quad (23)$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ and $\mathbf{R}$ is known. The observable output $\mathbf{y}_i^k$ of the FO model has a different dimension than the one of the PO model. Therefore, we distinguish between $\mathbf{R} = \mathbf{R}_{FO}$ and $\mathbf{R} = \mathbf{R}_{PO}$. Details about what time grid is used for each model, as well as the matrices $\mathbf{R}$, are reported in Table III.

Once the set of labelled time series data $\mathbb{D}$ is obtained, this set is transformed into a set of labelled Maximum *A Posteriori* inferential model parameter estimates $\mathbb{D}_{\text{MAP}} = \{(\boldsymbol{\theta}_{\text{MAP}}^k, c^k)\}_k$. Note that since we have chosen a flat prior over R, $\boldsymbol{\theta}_{\text{MAP}}^k$ will be a maximum likelihood estimate constrained to R:

$$\boldsymbol{\theta}_{\text{MAP}}^k = \arg\max_{\boldsymbol{\theta} \in R} \Big\{ \log(p(\mathbf{Y}^k \mid \boldsymbol{\theta}, \mathbf{t}^k; \mathbf{R})) \Big\}. \quad (24)$$

We remark that the argmax operation does not determine $\boldsymbol{\theta}_{\text{MAP}}^k$ uniquely, due to structural unidentifiability of the system. The problem given in Eq. (24) is solved using MATLAB's `simulannealbnd` function which can be used for

TABLE II
GROUND TRUTH PARAMETER CONFIGURATIONS FOR BINARY CLASSIFICATION TASKS OF THE DIFFERENT MODELS.

System	N_{train}	N_{test}	$\boldsymbol{\mu}_0$	$\boldsymbol{\mu}_1$	Σ_0, Σ_1
Toy model	100	200	$(a, b) = (1, 1)$	$(a, b) = 0.9 \cdot (1, 1)$	$10^{-4} I_2$
CCM2	100	200	$(k_{01}, k_{02}, k_{12}, k_{21})$ $= (0.015, 0.015, 0.074, 0.01)$	$(k_{01}, k_{02}, k_{12}, k_{21})$ $= (0.015, 0.015, 0.059, 0.008)$	$10^{-7} I_4$
CCM4	800	1000	$k_{0i} = 0.015,$ $(k_{12}, k_{23}, k_{34}, k_{21}, k_{32}, k_{43}) =$ $10^{-2}(7.4, 1, 7.4, 1, 7.4, 1)$	$k_{0i} = 0.015,$ $(k_{12}, k_{23}, k_{34}, k_{21}, k_{32}, k_{43}) =$ $10^{-2}(3.7, 0.5, 3.7, 0.5, 3.7, 0.5)$	$10^{-7} I_{10}$
CML	800	1000	identical to CCM4	identical to CCM4	$10^{-7} I_{10}$
BR	200	400	$(b_1, b_2, \mu_m, K_s, Y, K_d)$ $= (1.25, 30, 0.5, 3, 0.6, 0.05)$	$(b_1, b_2, \mu_m, K_s, Y, K_d)$ $= (1.25, 30, 0.5, 3, 0.48, 0.05)$	$\text{diag}(10^{-2} v),$ $v = (1, 100, 10^{-2}, 1, 10^{-2}, 10^{-4})$

constrained optimization. The outcome of the data transformation process is the set $\mathbb{D}_{\text{MAP}}$ and its SIM counterpart $\mathbb{D}_{\text{SIM}} = \{(\boldsymbol{\Phi}^k, c^k)\}_k$, where each $\boldsymbol{\Phi}^k$ is obtained as described in Eq. (9). For the fully observed dynamical model, only $\mathbb{D}_{\text{MAP}}$ is generated, whereas for the partially observed (and thus unidentifiable) model, both $\mathbb{D}_{\text{MAP}}$ and $\mathbb{D}_{\text{SIM}}$ are produced.

Once the sets $\mathbb{D}_{\text{MAP}}$ and $\mathbb{D}_{\text{SIM}}$ have been created, Support Vector Machine (SVM) classifiers are trained to learn the binary classification rule. A separate hold-out test data set (never used for training) is employed to assess the generalisation performance. The number of training and test examples produced is also reported in Table II. As an increasing number of training examples are made available, the generalization error and the relative number of support vectors are recorded as a function of the number of training examples per class. For each number of available training examples per class, the classifier is trained on 30 randomly sub-sampled datasets, and the mean and standard deviation of the generalization error and relative number of support vectors are reported as a function of the number of training examples. Training the classifier for 30 independent trials permits the capture of the variability in classification performance for a given number of training examples while keeping the runtime of the experiments feasibly low.

For SVM training we use MATLAB's `fitcsvm` with a Gaussian kernel. The kernel scale and the hyperparameter governing the penalization of misclassification (`BoxConstraint`) are selected by means of 10-fold cross-validation for each round of classifier training.

2) Experiment 2: This experiment is designed to study the robustness of SIM with respect to observational noise. For this purpose only the PO model and the PO model + SIM are compared. The overall setup mirrors that of experiment 1 with a few key differences. First, the number of training examples per class is kept constant, while the amount of observational noise is varied. We achieve this by setting $\mathbf{R} = \sigma^2 I$, where σ is adjusted to correspond to different levels of signal-to-noise ratio (SNR). Since all time series in experiment 2 are univariate, we assume a given time series of outputs

$$y_i = h(\mathbf{x}_{t_i}) + \epsilon_{t_i}, \quad i = 1, \ldots, L, \tag{25}$$

with $\epsilon_{t_i} \sim \mathcal{N}(0, \sigma^2)$. Furthermore, the power of the observed signal is defined as:

$$P_{\text{signal}} = \frac{1}{t_L - t_1} \int_{t_1}^{t_L} h(\tau)^2 d\tau, \tag{26}$$

with the power of the noise given by $P_{\text{noise}} = \sigma^2$ [51]. The SNR is the ratio: $SNR = P_{\text{signal}}/P_{\text{noise}}$, where the range $1 \leq \text{SNR} \leq 10^3$ is considered. Changes in the observational noise are applied to both training and test data. For each value of σ, the classifier is evaluated on 50 randomly sub-sampled datasets. The mean and standard deviation of the generalization error, as well as relative number of support vectors, are reported. Additionally, the following three quantities are computed for each example model. 1) $\Delta\epsilon^*$ (the maximum difference in mean generalization error), 2) SNR^* (the SNR at which that maximum occurs), and 3) $\langle \Delta\epsilon \rangle$ denoting the average of the difference between the generalization error curves obtained for the PO model and the PO model + SIM.

3) Experiment 3: This experiment is designed to study the robustness of SIM with respect to sparsity and irregularity of the time series data. Again, the overall setup is identical to that of experiment 1. In contrast to experiment 2, the observational noise is fixed. Time series are generated on three different types of time grids: A dense grid t_{dense}, which corresponds to frequent and regular measurements; a sparse grid t_{sparse}, which is regular like t_{dense} but only contains 40% of the points, and irregular grids t^k_{irr} containing sparse and irregular measurements different for every observation k. Each irregular time grid contains between 25% and 40% of the number of points in t_{dense}. Time points in the irregular grid are sampled uniformly at random between the first and last time points in t_{dense}. Notably, all time grids contain $t = 0$. The choices for t_{dense} for the different example models are reported in Table II. The configurations of t_{sparse} and t^k_{irr} follow from the choices of t_{dense}. In addition to the PO model and PO model + SIM, we also considered classification by means of a Multi-Layer Perceptron (MLP), which is not model-based and required some additional strategy to deal with the time series on the irregular grids. To this end we used interpolation by means of Linear Regression and Gaussian Process models. While for the dense and sparse grids, the MLP methods tend to perform better, we observed catastrophic deterioration of the MLP performance on the irregular grids for most of the example models, leading us to not pursue this approach any further. Finally, for experiment 3, the mean and standard deviation of the generalization error are reported for the three different types of time grids.

4) Experiment 4: To validate SIM on a real-world dataset we use the prednisone conversion model and dataset studied in [45]. Time series taken from 12 participants are found to group

TABLE III
EXPERIMENTAL CONFIGURATIONS FOR PARAMETERS RELATED TO TIME GRIDS AND OBSERVATIONAL NOISE.

System	t_{dense}	$\mathbf{R}_{FO}$	$\mathbf{R}_{PO}$
Toy model	$0, 0.1, \ldots, 1.0$	0.01	-
CCM2	$0, 10, \ldots, 240$	$10^2\mathbf{I}_2$	10^2
CCM4	$0, 10, \ldots, 240$	$10^2\mathbf{I}_4$	10^2
CML	$0, 10, \ldots, 240$	$10^2\mathbf{I}_4$	10^2
BR	$0, 1, \ldots, 12$	$\mathbf{I}_2$	1

into three classes: *slow absorbers*, *medium absorbers* and *fast absorbers* of prednisone. The dataset is imbalanced: there are 2 slow absorbers, 5 medium absorbers and 5 fast absorbers. The time series have varying number of measurements, ranging from 5 to 12 observed time points. To increase the degree of data sparsity every 2nd measurement is removed for demonstration. For the multi-class classification problem we deploy an Error Correcting Output Codes (ECOC) model with Support Vector Machines (SVM) as binary learners[2]. We use leave-one-out cross-validation to estimate the out-of-sample performance and employ the SVMs hyper-parameter tuning as described in experiment 1. We compare the PO model with and without SIM and summarise results in Figure 8.

IV. RESULTS

In the following, the results of experiments 1, 2 and 3 are presented in detail for the batch reactor model example. The results for all other example models are qualitatively similar and summarized in Table IV, V and VI. The results for the other example models are presented in detail in Appendix D.

A. Experiment 1

The outcomes of experiment 1 for the batch reactor model are summarized in Figure 5. Comparing the training outcomes of the fully observed (FO) dynamical model to the partially observed (PO) dynamical model, the results are not surprising. The training data obtained for the FO model are a super-set of the data available for the PO model. One would therefore expect that the classifier training with the FO model is more successful than training with the PO model. This is indeed reflected in Figure 5. For any amount of training data available, the FO curve for the generalisation error lies significantly below the PO curve. The same is true for the relative number of support vectors. As expected, using training data which include observations from all compartments, the problem of structural identifiability does not arise and it is possible to achieve better classification performance by fitting models of relatively low complexity.

The outcomes become more interesting when comparing the performance of the FO and PO models to those for the PO model where SIM was applied. Considering the generalization error, it is clear that the PO model + SIM outperforms the PO model consistently when the number of training examples is less than 50. Subsequently, the PO model and PO model + SIM reach comparable levels of generalisation error. Neither the PO model nor the PO model + SIM quite reach the performance

[2]Implemented in MATLAB's `fitcecoc`.

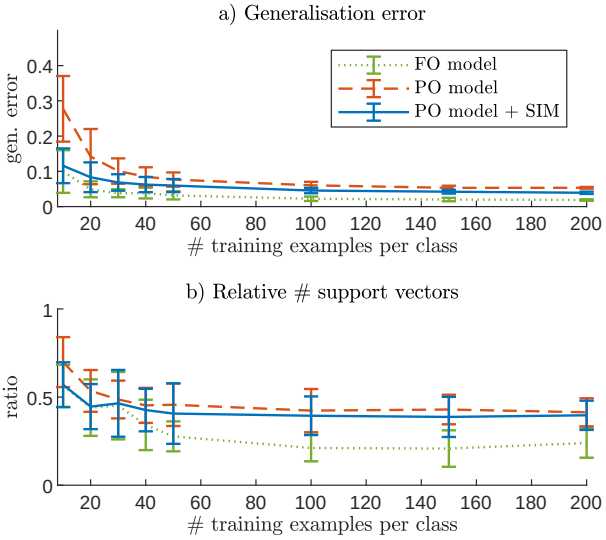

Fig. 5. Experiment 1 showing improved classification with partially observed batch reactor model due to SIM. Displayed are learning curves obtained from classifier training based on the fully observed (FO) dynamical model (dotted green) and the partially observed (PO) dynamical model, with and without application of SIM (marked with solid blue and dashed orange curves, respectively). The training and test data used were generated on the dense time grid t_{dense} with fixed observational noise $\sigma = 1$ on each observed component.

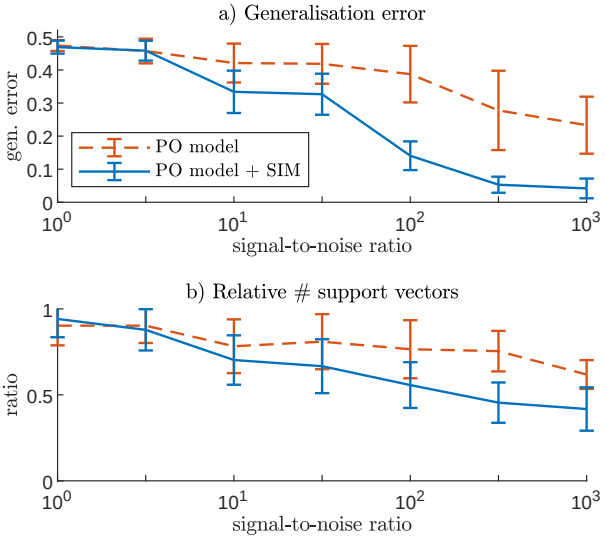

Fig. 6. Experiment 2 for the partially observed batch reactor model showing that SIM is robust to observational noise. Displayed are generalization error and relative number of support vectors, each as a function of the observational noise. Classification performance is compared for the partially observed (PO) dynamical model, with and without application of SIM (marked with solid blue and dashed orange curves, respectively). Training and test data are generated on the dense time grid t_{dense} with $N_{\text{train}} = 20$ and $N_{\text{test}} = 400$.

of the FO model. A similar situation can be observed for the relative number of support vectors. Up to 50 training examples, the mean curve for the PO model + SIM is very similar to the mean curve for the FO model. After 50 training examples, the mean curve for the PO model + SIM becomes evermore similar to that for the PO model. The FO, PO and PO + SIM curves obtained for the number of support vectors in Figure 5 are overall very similar to one another (when accounting for the observed standard deviations) and the effect of SIM is

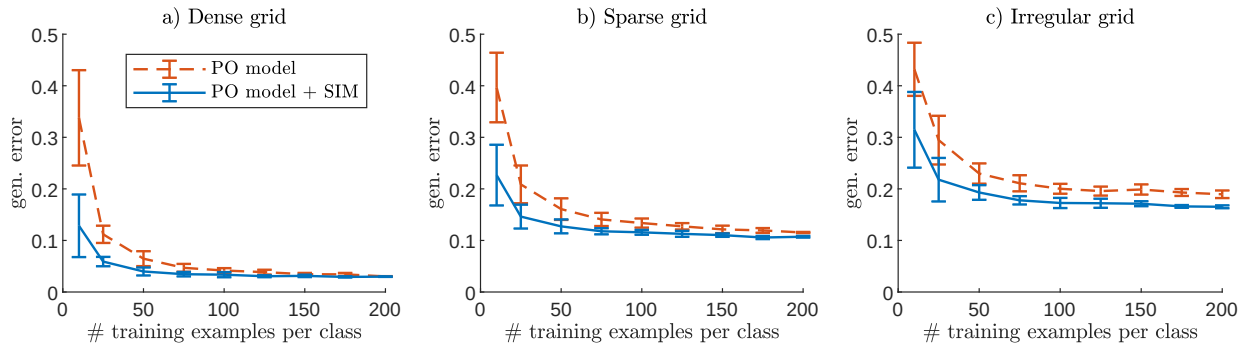

Fig. 7. Experiment 3 for the partially observed batch reactor model showing SIM is robust to changes in regularity and sparsity of the observed time series data. Displayed are the learning curves obtained from classifier training for the partially observed (PO) dynamical model, with and without application of SIM (marked with solid blue and dashed orange curves, respectively). The training and test data used were generated on the three different time grids t_{dense}, t_{sparse} and t_{irr}, displayed in the left, middle and right panel, respectively. Observational noise is fixed at $\sigma = 1$ on each observed component.

TABLE IV
SUMMARY OF EXPERIMENT 1 COMPARING CLASSIFICATION WITH THE FULLY OBSERVED (FO) MODEL, PARTIALLY OBSERVED (PO) MODEL AND PARTIALLY OBSERVED MODEL WITH SIM (PO + SIM). MEAN GENERALIZATION ERRORS (AND STANDARD DEVIATIONS), EVALUATED AT THE LOWEST NUMBER OF TRAINING EXAMPLES, ARE SHOWN FOR ALL EXAMPLE SYSTEMS.

	Examples		Generalization error at $N_{\min}$			Generalization error at $N_{\max}$		
System	$N_{\min}$	$N_{\max}$	FO	PO	PO + SIM	FO	PO	PO + SIM
CCM2	10	100	.0236 (.0304)	.0734 (.0748)	.0336 (.0230)	.0004 (.0009)	.0042 (.0012)	.0022 (.0016)
CCM4	10	400	.0568 (.0430)	.3326 (.0484)	.1855 (.0670)	.0007 (.0007)	.0174 (.0040)	.0133 (.0018)
CML	10	400	.0707 (.0645)	.3705 (.0516)	.1923 (.0648)	.0003 (.0005)	.0112 (.0012)	.0098 (.0010)
BR	10	200	.0995 (.0605)	.2774 (.0931)	.1158 (.0493)	.0187 (.0022)	.0534 (.0024)	.0392 (.0033)

TABLE V
SUMMARY OF EXPERIMENT 2. MAXIMUM ($\Delta\epsilon^*$) AND MEAN ($\langle\Delta\epsilon\rangle$) DIFFERENCE IN GENERALIZATION ERROR DUE TO SIM. THE SNR AT WHICH $\Delta\epsilon^*$ OCCURS IS SNR*. VALUES OF SNR* ARE IN log-SCALE.

System	N_{train}	SNR*	$\Delta\epsilon^*$	$\langle\Delta\epsilon\rangle$
Toy model	10	2.5	.16	.07
CCM2	10	2.5	.03	.02
CCM4	10	4.0	.20	.09
CML	10	3.5	.22	.09
BR	10	2.0	.23	.13

TABLE VI
SUMMARY OF EXPERIMENT 3: MEAN GENERALIZATION ERRORS (STDS) OF SIM WITH DENSE, SPARSE AND IRREGULAR OBSERVATION GRIDS AND $N_{\min} = 10$ TRAINING SAMPLES FOR EACH MODEL.

Setting \ System		Toy	CCM2	CCM4	CML	BR
t_{dense}	PO	.03 (.04)	.05 (.04)	.33 (.04)	.33 (.05)	.34 (.09)
	PO+SIM	.00 (.00)	.02 (.01)	.16 (.07)	.18 (.05)	.13 (.06)
t_{sparse}	PO	.05 (.04)	.17 (.06)	.41 (.03)	.42 (.03)	.40 (.07)
	PO+SIM	.00 (.00)	.09 (.03)	.27 (.05)	.27 (.06)	.23 (.06)
t_{irr}	PO	.06 (.06)	.19 (.07)	.40 (.03)	.42 (.03)	.43 (.05)
	PO+SIM	.00 (.00)	.17 (.04)	.25 (.06)	.28 (.04)	.31 (.07)

less clearly visible. However, in the low data regime up to 50 examples, SIM turns the classification problem into one with a less complex decision boundary associated with fewer support vectors and reduced generalisation error.

The outcomes of experiment 1 for the other models are summarized in Table IV and provide a similar picture. When comparing the generalization errors at the maximal number of training examples $N_{\max}$, the values for the PO model and the PO model + SIM are typically very similar with the values of the FO model, being significantly lower. However, at the minimal number of training examples, $N_{\min}$, the PO model is clearly outperformed by the PO model + SIM, which, in turn, is clearly outperformed by the FO model.

The conclusion to be drawn is clear: for densely sampled time series data and with relatively low observational noise present, SIM approach significantly reduces the complexity of the classification problem. Notably, when relatively little training data are available, the classification performance is remarkably close to the performance that would be attained if the underlying dynamical model was fully observed. Utilizing the information from the structural identifiability analysis therefore exhibits a good alternative, when certain measurements are unobtainable, for improving machine learning performance, in particular when training data are limited.

B. Experiment 2

The outcomes of experiment 2 for the batch reactor model are summarized in Figure 6. Several trends emerge from the results. As the SNR increases, both the generalization error and relative number of support vectors decrease, as the classification problem becomes easier with less noise. When the SNR approaches 1 distinguishing the classes becomes more and more difficult and the generalization error approaches 0.5 due to the signal degradation. Nevertheless, for a wide range of SNR values, the generalization error is significantly reduced by utilizing SIM. Similarly, using SIM requires fewer support vectors with higher SNR, and more training samples near the boundary become assigned as the SNR decreases. Table V summarizes the experimental results for the remaining example models. The maximal error difference $\Delta\epsilon^*$ occurs at

high SNR values, as expected, since structural identifiability (in contrast to practical identifiability) describes an inherent model property, that is independent of observational noise, making the SIM most effective at zero noise (SNR $\to \infty$). The mean difference in the generalization error $\langle \Delta\epsilon \rangle$ is positive for all example models, indicating a reduction in error due to SIM across various noise levels. In conclusion, experiment 2 shows that SIM improves classification performance across a wide range of observational noise levels, with the benefits growing as the difficulty increases.

C. Experiment 3

The outcomes of experiment 3 for the batch reactor model are summarized in Figure 7. It is to be noted that the presentation of the results for experiment 3 differ from those for experiments 1 and 2 in that for this experiment we plot *only* the generalization error. The learning curves are qualitatively the same as those presented in Figure 5 and the general trend is again clearly visible. Using time series data with observations in t_{dense} yields better generalization errors than training with data that are generated in t_{sparse}. Similarly, training with data generated in t_{sparse} yields better overall results than training on data that are generated in t_{irr}. This is reasonable, since the data in t_{dense} simply contain more information than those in t_{sparse} and t_{irr}. The effect of SIM appears to be robust with respect to the time grid used: for each time grid the application of SIM yields reduced generalization errors. Since the difference between the PO model and the PO model + SIM are most pronounced for relatively small amounts of training data, Table VI summarizes the outcomes of experiment 3 for all example models at the minimal number of training examples $N_{\min}$. In all cases, the application of SIM leads to a reduction in average generalization error.

D. Experiment 4

As shown in Figure 8 SIM clearly improves the classification performance of the medium and fast absorbers: two examples which are misclassified under the PO model are correctly classified when SIM is used. The 2 slow absorbers cannot be retrieved correctly due to the class imbalance.

V. Discussion

The computational complexity of SIM breaks down into three components: the SI analysis, MAP estimation and machine learning training. Each of these may be considered independent and the exact choice of algorithm to tackle each step should be tailored to the application at hand. A standard training pipeline for a model-based approach would involve finding suitable representations of data in the model space (MAP estimation in this paper) and subsequent training of the classifier of choice. The preceding SIM analysis is performed once, but its complexity strongly depends on the problem at hand. Due to these factors and involvement of symbolic manipulation, the computational complexity of SI analysis methods is generally difficult to determine. Consequently, existing approaches are primarily assessed through empirical comparisons, as illustrated in the recent benchmarking study [38] covering a range of SI analysis methods.

It was observed that the SIM approach is most effective for relatively small amounts of training data. This is because SIM removes redundancies in the space of the original model parameters and thus makes the decision boundary in the space of identifiable parameter combinations simpler (cf. Figure 1). As the amount of available data increases, the effect of SIM is diminished since the additional data now suffice to resolve the class-membership distribution in the space of the original parameters. This means that SIM has a regularizing effect on the classifier training and is particularly useful whenever there are relatively few data available, which is common in biomedical applications.

Deep Learning learning methods have been applied to the problems of time series analysis and classification with great success. However, with the large number of weights to be trained, deep networks have a tendency to over-fit and effective regularization becomes a strict necessity when working in the small-data regime. Recently, Physics-informed Neural Networks (PINN) have been introduced which regularize the training process by the incorporation of any physical laws in the form of ODEs and/or partial differential equations (PDE) [52]. In this context, PINN can also be used for parameter estimation for ODEs (as a special case of PDEs). However, if a PINN were to be set up incorporating an ODE with unidentifiable parameters, then any form of parameter estimation would again become meaningless. A thorough Structural Identifiability analysis of the underlying dynamical model is therefore strongly recommended when employing a PINN for parameter estimation.

In any situation involving high-stakes decision making, interpretability is of critical importance. A recent review, in which 9 state-of-the-art deep learning methods for time series classification are compared, found that only 2 out of the 9 methods studied address the issue of interpreting the decision taken by the neural network [53]. Using SIM, even though a given classifier is trained on data in the space of identifiable parameter combinations, the learned decision boundary can be recovered in the space of the original model parameters. This makes the learned decision boundary interpretable for domain experts and increases trust in the trained model. Another example in which insight is generated from an unidentifiable model in a similar manner can be found in Bunte et al. [45]. SIM not only improves classification performance but also preserves interpretability of the model-based approach.

There are a number of limitations to be considered when applying the SIM approach. For one, the extent to which the existence of non-trivial output-equivalent manifolds of models actually hampers classification performance is hard to predict a priori. Depending on the optimization scheme employed to maximize the log-likelihood function, and depending on the dynamical system in question, performance degradation may be more or less severe, making the effectiveness of SIM situation-dependent. Moreover, in [7], the authors point out that working with point estimates (like MAP) to represent time series data in the parameter space of a given dynamical model

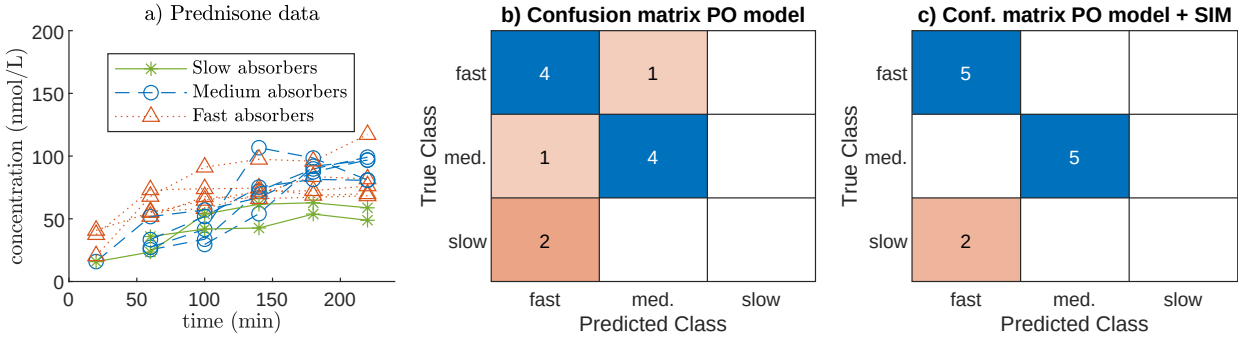

Fig. 8. Experiment 4 validates SIM on a three-class problem of prednisone conversion with classification measurements shown in panel a). The confusion matrix resulting from leave-one-out cross-validation based on the PO model without and with SIM are depicted in panel b (error 0.33) and c (error 0.17).

comes with inherent difficulties because such estimates do not quantify the uncertainty for models *around* these estimates. As an alternative, the authors propose a fully Bayesian approach and represent each time series observation as a posterior distribution over the entire model parameter space.

This paper extends beyond previous work [7], [45] by incorporating structural identifiability (SI) analysis even for structurally non-identifiable (SNI) models. While we build on the "learning in the model space" (LiMS) framework in [7], structural identifiability is not addressed there. The model-based clustering approach in [45] includes SI analysis, but the model is structurally *locally* identifiable with finitely many solutions, that allowed the exclusion of biologically unfeasible cases. In contrast, the present work makes use of SI analysis regardless of whether the model is locally identifiable, enabling us to handle structurally non-identifiable (SNI) models and thus go beyond the scope of [7] and [45].

TABLE VII
SI ANALYSIS AND SNI MODELS IN RELATED WORKS.

Content \ Paper	Shen et al. [7]	Bunte et al. [45]	present work
SI analysis	✗	✓	✓
Models are SNI	✗	✗	✓

VI. CONCLUSION

Model-based approaches for time series classification can be effectively utilized when a model of the underlying dynamical process is available [7]. Using structural identifiability (SI) analysis, structurally identifiable parameter combinations of the dynamical model can be obtained. Individual time series observations may then be represented as point estimates in the original parameter space or in the space of structurally identifiable parameter combinations. We introduced a strategy dubbed **S**tructural-**I**dentifiability **M**apping (SIM) and demonstrated that SIM improves classification performance for time series data when taking a model-based approach and the underlying dynamical model is structurally unidentifiable.

It has been shown on a set of relevant example systems that classification performance is significantly improved when learning with data represented in the space of structurally identifiable parameter combinations. The increase in performance also persists when time series data of varying quality are produced: for all types of time grids (dense, sparse and irregular) as well as for varying levels of the observational noise introduced, learning in the space of structurally identifiable parameter combinations outperforms learning in the original parameter space. This work demonstrates incorporating SI analysis directly into the learning process for classification. SIM is straightforward and can be applied whenever a SI analysis can be carried out. An explicit reparametrisation of a given dynamical model in terms of fewer, structurally identifiable parameters is not needed in order to benefit from SI analysis. This is especially important when explicit expressions for structurally identifiable parameter combinations are available, but suitable model reparametrizations are not possible.

Finally, outcomes of the learning process stay interpretable: while interpretation in the space of structurally identifiable parameter combinations is not straightforward, any insight in this space may be translated back to the space of the original model parameters $g^{-1}(\Phi)$ which, in turn, are meaningful in the domain-specific context, for example stimulating mathematical modellers to formulate further viable constraints on system configurations, or simulate interventions to find the most effective or least invasive options.

REFERENCES

[1] K. Ø. Mikalsen, F. M. Bianchi, C. Soguero-Ruiz, S. O. Skrøvseth, R.-O. Lindsetmo, A. Revhaug, and R. Jenssen, "Learning similarities between irregularly sampled short multivariate time series from ehrs," *Proc. 3rd Int. Workshop Pattern Recognit. Healthcare Anal.*, pp. 1–6, 12 2016.

[2] E. Peter Carden and J. M. Brownjohn, "ARMA modelled time-series classification for structural health monitoring of civil infrastructure," *Mech. Syst. Signal Process.*, vol. 22, no. 2, pp. 295–314, 2008.

[3] E. Trentin, S. Scherer, and F. Schwenker, "Emotion recognition from speech signals via a probabilistic echo-state network," *Pattern Recognition Letters*, vol. 66, pp. 4–12, 2015.

[4] W. Pei, H. Dibeklioğlu, D. M. J. Tax, and L. van der Maaten, "Multivariate time-series classification using the hidden-unit logistic model," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 4, pp. 920–931, 2018.

[5] F. M. Bianchi, S. Scardapane, S. Løkse, and R. Jenssen, "Reservoir computing approaches for representation and classification of multivariate time series," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 5, pp. 2169–2179, 2021.

[6] T. Bradde, G. Fracastoro, and G. C. Calafiore, "Multiclass sparse centroids with application to fast time series classification," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–6, 2021.

[7] Y. Shen, P. Tino, and K. Tsaneva-Atanasova, "Classification framework for partially observed dynamical systems," *Physical Review E*, vol. 95, no. 4, p. 043303, 2017.

[8] C. Meng, S. Griesemer, D. Cao, S. Seo, and Y. Liu, "When physics meets machine learning: a survey of physics-informed machine learning," *Machine Learning for Computational Science and Engineering*, vol. 1, p. 20, 5 2025.

[9] Y. Zheng, H.-T. Zhang, Z. Yue, and J. Wang, "Sparse Bayesian Learning for Switching Network Identification," *IEEE Trans. on Cybernetics*, vol. 54, pp. 7642–7655, Dec. 2024.

[10] A. K. Kamath, S. G. Anavatti, and M. Feroskhan, "A Physics-Informed Neural Network Approach to Augmented Dynamics Visual Servoing of Multirotors," *IEEE Trans. on Cybernetics*, vol. 54, pp. 6319–6332, Nov. 2024.

[11] X. Ping, X. Luan, S. Zhao, F. Ding, and F. Liu, "Parameter Transfer Identification for Nonidentical Dynamic Systems Using Variational Inference," *IEEE Trans. on Systems, Man, and Cybernetics: Systems*, vol. 55, pp. 712–720, Jan. 2025.

[12] Z. Yuan, W. V. Ransbeeck, G. A. Wiggins, and D. Botteldooren, "A Dynamic Systems Approach to Modeling Human–Machine Rhythm Interaction," *IEEE Trans. on Cybernetics*, vol. 55, pp. 2052–2064, May 2025.

[13] N. Ahmadi, Q. Cao, J. D. Humphrey, and G. E. Karniadakis, "Physics-informed machine learning in biomedical science and engineering," 2025.

[14] A. Krishnapriyan, A. Gholami, S. Zhe, R. Kirby, and M. W. Mahoney, "Characterizing possible failure modes in physics-informed neural networks," in *Advances in Neural Information Processing Systems* (M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, eds.), vol. 34, pp. 26548–26560, Curran Associates, Inc., 2021.

[15] M. Burger and S. Kabri, "Learned Regularization for Inverse Problems: Insights from a Spectral Model," Dec. 2023.

[16] C. Balsells-Rodas, Y. Wang, and Y. Li, "On the identifiability of switching dynamical systems," in *Proc. 41st Int. Conf. Mach. Learn.* (R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, and F. Berkenkamp, eds.), vol. 235, pp. 2639–2672, PMLR, 2024.

[17] R. Hermann and A. Krener, "Nonlinear controllability and observability," *IEEE Trans. Automat. Contr.*, vol. 22, no. 5, pp. 728–740, 1977.

[18] A. F. Villaverde, A. Barreiro, and A. Papachristodoulou, "Structural Identifiability of Dynamic Systems Biology Models," *PLOS Computational Biology*, vol. 12, no. 10, p. e1005153, 2016.

[19] S. C. Hora, "Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management," *Reliability Engineering & System Safety*, vol. 54, no. 2-3, pp. 217–223, 1996.

[20] A. D. Kiureghian and O. Ditlevsen, "Aleatory or epistemic? does it matter?," *Structural Safety*, vol. 31, no. 2, pp. 105–112, 2009. Risk Acceptance and Risk Communication.

[21] E. Hüllermeier and W. Waegeman, "Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods," *Machine Learning*, vol. 110, pp. 457–506, 2021.

[22] F. Anstett-Collin, L. Denis-Vidal, and G. Millérioux, "A priori identifiability: An overview on definitions and approaches," *Annual Reviews in Control*, vol. 50, pp. 139–149, 2020.

[23] R. Bellman and K. J. Åström, "On structural identifiability," *Mathematical biosciences*, vol. 7, no. 3-4, pp. 329–339, 1970.

[24] H. Pohjanpalo, "System identifiability based on the power series expansion of the solution," *Mathematical biosciences*, vol. 41, no. 1-2, pp. 21–33, 1978.

[25] E. Walter and L. Pronzato, "On the identifiability and distinguishability of nonlinear parametric models," *Mathematics and computers in simulation*, vol. 42, no. 2-3, pp. 125–134, 1996.

[26] E. Tunali and T.-J. Tarn, "New results for identifiability of nonlinear systems," *IEEE Trans. Automat. Contr.*, vol. 32, no. 2, pp. 146–154, 1987.

[27] S. Vajda and H. Rabitz, "State isomorphism approach to global identifiability of nonlinear systems," *IEEE Trans. Automat. Contr.*, vol. 34, no. 2, pp. 220–223, 1989.

[28] H. J. Sussmann, "Existence and uniqueness of minimal realizations of nonlinear systems," *Mathematical systems theory*, vol. 10, no. 1, pp. 263–284, 1976.

[29] S. Diop and M. Fliess, "Nonlinear observability, identifiability, and persistent trajectories," in *Proc. 30th IEEE Conf. Dec. Control.*, pp. 714–719, IEEE, 1991.

[30] L. Ljung and T. Glad, "On global identifiability for arbitrary model parametrizations," *Automatica*, vol. 30, no. 2, pp. 265–276, 1994.

[31] R. Jain, S. Narasimhan, and N. P. Bhatt, "A priori parameter identifiability in models with non-rational functions," *Automatica*, vol. 109, p. 108513, 2019.

[32] F.-G. Wieland, A. L. Hauber, M. Rosenblatt, C. Tönsing, and J. Timmer, "On structural and practical identifiability," *Current Opinion in Systems Biology*, vol. 25, pp. 60–69, 2021.

[33] I. Ilmer, A. Ovchinnikov, and G. Pogudin, "Web-based structural identifiability analyzer," in *Computational Methods in Systems Biology* (E. Cinquemani and L. Paulevé, eds.), (Cham), pp. 254–265, Springer International Publishing, 2021.

[34] R. Dong, C. Goodbrake, H. A. Harrington, and G. Pogudin, "Differential elimination for dynamical models via projections with applications to structural identifiability," *SIAM Journal on Applied Algebra and Geometry*, vol. 7, no. 1, pp. 194–235, 2023.

[35] G. Massonis, J. R. Banga, and A. F. Villaverde, "Autorepar: A method to obtain identifiable and observable reparameterizations of dynamic models with mechanistic insights," *Int. J. Robust Nonlinear Control.*, 2021.

[36] G. Massonis and A. F. Villaverde, "Finding and breaking lie symmetries: Implications for structural identifiability and observability in biological modelling," *Symmetry*, vol. 12, no. 3, 2020.

[37] N. Meshkat, C. E.-z. Kuo, and J. DiStefano III, "On finding and using identifiable parameter combinations in nonlinear dynamic systems biology models and combos: a novel web implementation," *PloS one*, vol. 9, no. 10, p. e110261, 2014.

[38] X. Rey Barreiro and A. F. Villaverde, "Benchmarking tools for a priori identifiability analysis," *Bioinformatics*, vol. 39, 01 2023. btad065.

[39] F. C. Coelho, C. T. Codeço, and M. G. M. Gomes, "A bayesian framework for parameter estimation in dynamical models," *PloS one*, vol. 6, no. 5, p. e19616, 2011.

[40] N. J. Linden, B. Kramer, and P. Rangamani, "Bayesian parameter estimation for dynamical models in systems biology," *PLOS Computational Biology*, vol. 18, no. 10, p. e1010651, 2022.

[41] J. A. Jacquez *et al.*, *Compartmental analysis in biology and medicine*. New York, Elsevier Pub. Co., 1972.

[42] C. M. Metzler, "Usefulness of the two-compartment open model in pharmacokinetics," *J. Am. Stat. Assoc.*, vol. 66, no. 333, pp. 49–53, 1971.

[43] J. E. Sager, J. Yu, I. Ragueneau-Majlessi, and N. Isoherranen, "Physiologically based pharmacokinetic (PBPK) modeling and simulation approaches: A systematic review of published models, applications, and model verification," *Drug Metab. Dispos.*, vol. 43, no. 11, pp. 1823–1837, 2015.

[44] B. C.-M. Chen, E. M. Landaw, and J. J. DiStefano, "Algorithms for the identifiable parameter combinations and parameter bounds of unidentifiable catenary compartmental models," *Mathematical Biosciences*, vol. 76, no. 1, pp. 59–68, 1985.

[45] K. Bunte, D. J. Smith, M. J. Chappell, Z. K. Hassan-Smith, J. W. Tomlinson, W. Arlt, and P. Tiňo, "Learning pharmacokinetic models for in vivo glucocorticoid activation," *Journal of Theoretical Biology*, vol. 455, pp. 222–231, 2018.

[46] N. Meshkat and S. Sullivant, "Identifiable reparametrizations of linear compartment models," *Journal of Symbolic Computation*, vol. 63, pp. 46–67, 2014.

[47] D. K. Button, "Kinetics of nutrient-limited transport and microbial growth," *Microbiological reviews*, vol. 49, no. 3, pp. 270–297, 1985.

[48] A. Holmberg, "On the practical identifiability of microbial growth models incorporating michaelis-menten type nonlinearities," *Mathematical Biosciences*, vol. 62, no. 1, pp. 23–43, 1982.

[49] M. J. Chappell and K. R. Godfrey, "Structural identifiability of the parameters of a nonlinear batch reactor model," *Mathematical Biosciences*, vol. 108, no. 2, pp. 241–251, 1992.

[50] N. D. Evans and M. J. Chappell, "Extensions to a procedure for generating locally identifiable reparameterisations of unidentifiable systems," *Mathematical biosciences*, vol. 168, no. 2, pp. 137–159, 2000.

[51] A. Papoulis, *Random variables and stochastic processes*. McGraw Hill, 1965.

[52] M. Raissi, P. Perdikaris, and G. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019.

[53] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Deep learning for time series classification: a review," *Data mining and knowledge discovery*, vol. 33, no. 4, pp. 917–963, 2019.