frontiers | Frontiers in Artificial Intelligence

TYPE Original Research
PUBLISHED 19 December 2025
DOI 10.3389/frai.2025.1706566

Retrieving interpretability to support vector machine regression models in dynamic system identification

Johan Pena-Campos[1,2]*, Diego Patino[2†],
Carlos Ocampo-Martinez[1†], Julio C. Ramos-Fernández[3†],
Margot Salas-Brown[4†] and Alexander Caicedo[2,5]

[1]Automatic Control Department (ESAII), Universitat Politécnica de Catalunya, Barcelona, Spain, [2]Department of Electronic Engineering, Pontificia Universidad Javeriana, Bogotá, Colombia, [3]Faculty of Mathematical and Natural Sciences, Universidad Distrital Francisco José de Caldas, Bogotá, Colombia, [4]School of Exact Sciences and Engineering, Universidad Sergio Arboleda, Bogotá, Colombia, [5]Ressolve S.A.S, Medelln, Colombia

Check for updates

OPEN ACCESS

EDITED BY
Joao Gama,
University of Porto, Portugal

REVIEWED BY
Taha Hussein Ali,
Salahaddin University, Iraq
Heyam Hayawi,
University of Mosul, Iraq
Federico Javier Gonzalez,
University Institute of Gran Rosario, Argentina

*CORRESPONDENCE
Johan Pena-Campos
✉ johan.sebastian.pena@upc.edu;
✉ pena.johan@javeriana.edu.co

†These authors have contributed equally to this work

RECEIVED 16 September 2025
REVISED 18 November 2025
ACCEPTED 25 November 2025
PUBLISHED 19 December 2025

CITATION
Pena-Campos J, Patino D, Ocampo-Martinez C, Ramos-Fernández JC, Salas-Brown M and Caicedo A (2025) Retrieving interpretability to support vector machine regression models in dynamic system identification.
Front. Artif. Intell. 8:1706566.
doi: 10.3389/frai.2025.1706566

Black-box models, particularly Support Vector Machines (SVM), are widely employed for identifying dynamic systems due to their high predictive accuracy; however, their inherent lack of transparency hinders the understanding of how individual input variables contribute to the system output. Consequently, retrieving interpretability from these complex models has become a critical challenge in the control and identification community. This paper proposes a *post-hoc* functional decomposition algorithm based on Non-linear Oblique Subspace Projections (NObSP). The method decomposes the output of an already identified SVM regression model into a sum of partial (non)linear dynamic contributions associated with each input regressor. By operating in the non-linear feature space, NObSP utilizes oblique projections to mitigate cross-contributions from correlated regressors. Furthermore, an efficient out-of-sample extension is introduced to improve scalability. Numerical simulations performed on benchmark Wiener and Hammerstein structures demonstrate that the proposed method effectively retrieves the underlying partial nonlinear dynamics of each sub-system. Additionally, the computational analysis confirms that the proposed extension reduces the arithmetic complexity from $\mathcal{O}\left(N^3\right)$ to $\mathcal{O}\left(Nd^2\right)$, where d is the number of support vectors. These findings indicate that NObSP is a robust geometric framework for interpreting non-linear dynamic models, offering a scalable solution that successfully decouples blended dynamics without sacrificing the predictive power of the black-box model.

KEYWORDS

interpretability, oblique projections, support vector machine, Hammerstein-Wiener models, dynamic systems, system identification

1 Introduction

Interpretability of Support Vector Machine (SVM) or Neural Networks (NN) models, examples of black-box models, is a field of study that has recently gained attention, especially for the significant advances of machine learning models and their inclusion in fields such as medicine and law (Barredo Arrieta et al., 2020). For physicians, the accuracy of classification models is as important as understanding why the models provide some results. The lack of understanding of the model performance diminishes the confidence of the specialist in its use, even more so when the model's output differs from the one expected by the specialist. When addressing interpretability of a machine learning model, the aim is to understand how the input parameters influence the model's output. Interpretability can be addressed in two ways: *model* and *instance* explanation approaches. The *local* or *instance* approach tries to explain a prediction for a specific instance, making it valid just for its vicinity. This approach must not be generalized (Burkart and Huber, 2021). In contrast, *global* or *model* interpretability aims to provide information about the model functionality on its whole using only the training data (Barredo Arrieta et al., 2020; Burkart and Huber, 2021).

The framework of interpretability/explainability is still wide and open. In Luckey et al. (2022), the authors establish an explainable artificial intelligence (XAI) pipeline, where the explanation and interpretation are processes within this pipeline. The goal of the explanation is to identify the most relevant features that influence the classifier decision, i.e., illustrate which input features contribute the most to producing a decision. In the next process, interpretation, the features previously identified are associated with the problem-specific domain, i.e., mapping an abstract concept into a domain that makes sense to humans. Further, in Barredo Arrieta et al. (2020) the authors define interpretability as a passive characteristic of a model, part of its design, at the level that it has a sense for a human observer; and explicability as the active characteristic that clarify or detail the internal functioning in a model. In Burkart and Huber (2021), the authors address three concepts: (i) the interpretable models, which are entirely understandable and are built naturally or by using design principles; (ii) the approach of fitting a surrogate model that approximates a black box through local or global interpretable models, and (iii) the process of generating a local or global explanation. In addition, the development of interpretable models can be divided into two groups, according to the moment when interpretability/explainability is applied. If the explanation is produced ante-hoc, they can be called Interpretable (Burkart and Huber, 2021) or Transparent models (Barredo Arrieta et al., 2020; Luckey et al., 2022) whilst explainable models with *post-hoc* explanations.

Linear regression is considered a white box or an interpretable model since it is possible to know how each input variable has contributed to the output. White-box models are characterized for having *ante-hoc* interpretability, which means that they are interpretable on their own (Burkart and Huber, 2021). However, linear regression might lack accuracy in its predictions since it is not able to model nonlinearities that are not explicitly defined in the model design (Harrell et al., 2001). In contrast, black-box models have high accuracy and can adapt to nonlinearities but lack interpretability. For black-box models, *post-hoc* interpretability can be achieved through a global model-agnostic explanator (Barredo Arrieta et al., 2020), which is a white-box surrogate model that simulates the behavior of the black-box model (Burkart and Huber, 2021).

In recent years, powerful *post-hoc* and model-agnostic frameworks have become foundational to XAI. Notably, Local Interpretable Model-agnostic Explanations (LIME) (Ribeiro et al., 2016) and SHapley Additive exPlanations (SHAP) (Lundberg and Lee, 2017) are designed to explain the predictions of any classifier or regressor. These methods are highly effective for feature attribution, providing importance scores that quantify the contribution of each individual feature to a specific prediction.

However, applying these methods directly to dynamic systems is non-trivial, as standard implementations (e.g., KernelSHAP) often assume feature independence, which is fundamentally violated by the temporal correlations (autocorrelation) inherent in time-series data. Although adaptations for time-series have been proposed (Jutte et al., 2025; Sen et al., 2025; Theissler et al., 2022), their objective remains providing saliency scores (e.g., the importance of $u[n-k]$ at a specific time step k) rather than retrieving a complete functional dynamic.

Other families of XAI methods face similar limitations in this context. Gradient-based methods, such as Grad-CAM, have been adapted for 1D signals (Selvaraju et al., 2016; Aquino et al., 2022), but they are (i) model-specific to neural networks, requiring access to gradients and internal feature maps, and thus inapplicable to kernel-based models like SVMs, and (ii) focused on identifying saliency (i.e., which parts of the input were most critical), not decomposing the output.

Furthermore, visualization methods like Partial Dependence Plots (PDP) and Individual Conditional Expectation (ICE) plots, which do attempt to show a functional relationship, also struggle with dynamic systems. Their reliance on marginalizing features fails when strong correlations exist—as they always do between lagged regressors ($u[n-1]$, $u[n-2]$, etc.). This fact can lead to averaging over "impossible" regions of the feature space, yielding unreliable results (Rojat et al., 2021; Angelini et al., 2023; Shi et al., 2023).

This work, in contrast, addresses a different objective. The goal of the NObSP extension is not feature attribution but functional decomposition. The methodology leverages the specific geometric properties of kernel methods to reconstruct the entire partial (non)linear dynamic contribution of each input regressor (i.e., $\hat{y}_l$) as an additive component of the total system output. As demonstrated with the Wiener and Hammerstein examples, this decompositional approach, built on oblique projections specifically designed to handle correlated regressors, allows for the retrieval of the underlying sub-system dynamics—a different and more holistic form of interpretation than feature attribution or saliency mapping.

Considering nonlinearities, additive models can decompose the output of a nonlinear regression as the sum of the partial (non)linear contributions of each input variable and their interaction effects. In this context, some approaches already reported in the literature are sparse additive models (Ravikumar et al., 2009), functional ANOVA models (Abramovich and Angelini, 2006), and neural additive models (Agarwal et al., 2021). However, these methods require to specify, a priori, which are the most relevant input variables and interaction effects of interest. This restriction, imposed during the model definition, conditions the functionality of the methods. In contrast, black-box models, such as SVM and NN, do not need to define the variables of interest a priori, but they use all the available input-output observations to find the model that better fits the data. These black-box models generally have greater accuracy in their predictions but lack interpretability (DeVore et al., 2011). Nevertheless, in Rudin (2019), the authors express that it is a myth that there is necessarily a trade-off between accuracy and interpretability. Additionally, the authors mention that there exists a widespread belief that more complex models (black box) are more accurate, which often is not true, especially when the data are structured with a suitable representation in terms of naturally meaningful features.

SVM is a popular non-parametric framework that uses input data and their targets to estimate a model, which can be employed to generate predictions on unseen data (Vapnik, 1999). SVM produces a *black-box* model that fits the data but does not facilitate the interpretation of the results. As in Caicedo et al. (2019), in this article, the interpretability is understood as "the property of a model to express the output into additive terms of the partial (non)linear contributions of the input variables." Interpretability of Least-Squares Support Vector Machine (LS-SVM) has been addressed by employing a truncated multinomial expansion for classifiers (Van Belle and Lisboa, 2014), and using oblique subspace projections for regression models (Caicedo et al., 2019). In Ravikumar et al. (2009) and Abramovich and Angelini (2006), the authors propose to retrieve the interpretability of SVM forcing the model to be adjusted from prior knowledge, identifying the contributions of each input variable main effects and interaction effects. Here, the designer needs to define a priori, which input variables and interaction effects of interest are essential for the model. Other approaches use a geometric framework, decomposing the estimated observation vector as a linear additive term through oblique subspace projections (Bring, 1996). This approach is similar to the one proposed by Caicedo et al. (2019), where they use a nonlinear extension to oblique subspace projections (NObSP). Here, they used a static LS-SVM regression and considered that in the dual space, i.e., the transform space, the underlying model that relates the input and output variables is linear. Therefore, in this dual space the model can be decomposed into additive components. In Caicedo et al. (2019), they proposed to generate a basis for each subspace of interest using appropriate kernel evaluations, i.e., subspaces that span the (non)linear transformation of each input variable and their interaction effects. They demonstrate the use of NObSP through toy examples and showcase its application in the manufacturing industry using data from the compressive strength dataset from the University of California, Irvine (UCI) machine learning repository (Yeh, 1998). NObSP retrieves the functional relationships between the input and output variables for a static regression model using LS-SVM, even in the presence of correlated inputs. In addition, it does not require for the designer to define a priori the input variables and the interaction effects of interest.

While the previous discussion centered on static regression, the utility of black-box models extends prominently to the identification of nonlinear dynamic systems. In this context, Gonzalez-Olvera and Tang (2010) have proposed recurrent NN to identify dynamic systems, and Forgione et al. (2023) proposed a methodology to adapt the identified model using recurrent NN to the changes present in a non-stationary nonlinear system. Moreover, Yazdani et al. (2020) used deep learning algorithms to estimate the parameters of a differential equation that models a biological system. Likewise, Candon et al. (2022) compared the performance of multiple-input-single-output (MISO) system identification from linear regression models, Artificial NN, and deep learning strategies in the prediction of the representative bending and torsional load spectra on an aircraft wing based on strain sensors. Besides, Resendiz-Trejo et al. (2006) used a recursive SVM for MISO nonlinear system online identification, improving computational cost compared to SVM, while Espinoza et al. (2005) used a partially linear model with an LS-SVM to identify a combined linear-nonlinear model, with fewer parameters, better generalization ability and performance than a full nonlinear black-box model. In addition, Li J. et al. (2022) and Zong et al. (2021) considered block-oriented system identification approaches, where the nonlinear system is represented as an interconnection of linear and nonlinear blocks. Some examples of these nonlinear systems structures are a Wiener system, i.e., a linear block followed by a nonlinear block, and a Hammerstein system, i.e., a nonlinearity followed by a linear block (Li J. et al., 2022). In this context, some methodologies have been developed for the use of SVM and LS-SVM for the identification of Wiener systems (Castro-Garcia and Suykens, 2016; Bottegal et al., 2018; Bottegai et al., 2017), Hammerstein-Wiener systems (Goethals et al., 2005), and Wiener-Hammerstein systems (Falck et al., 2009). More recent works on these block structures use Particle Swarm Optimization (PSO) to obtain the parameters of the Hammerstein-Wiener nonlinear system, including the time delay (Li J. et al., 2022), and to identify the model in the presence of scarce measurements (Zong et al., 2021). In (Li F. et al., 2022) and Li et al. (2023c), the authors present a decouple identification scheme model for nonlinear systems through a structure of a Hammerstein system based on a neural fuzzy network and autoregressive exogenous (ARX) model. Finally, this methodology is analyzed with output noise (Li et al., 2023b), and extended for Hammerstein-Wiener (Li et al., 2023a) and Wiener (Li et al., 2024) structures.

Industrial processes like thermal, biological fermentation, chemical processes, pumped-storage power generating systems, and solar-wind hybrid power systems, among others, present nonlinear characteristics. Although the block structure of the Wiener and Hammerstein models is well-known in the literature, their structure helps to reflect the behavior of these types of systems integrating the linear dynamic model with a static memoryless nonlinear model (Li J. et al., 2022; Zong et al., 2021). In addition, time delay phenomenon is also encountered in metallurgy, refining, and glass industries with complex production links, increasing the adjustment time (Li J. et al., 2022).

In the presence of a MISO system, it will be advantageous to fit a general model to the system dynamics and then decompose its output into additive terms, where each term represents the output of a Wiener, Hammerstein, or Wiener-Hammerstein system. Here, a strategy such as NObSP might be of help. However, NObSP has yet to be developed for dynamic systems identification. In addition, to decompose the output of the model using NObSP, it is necessary to compute oblique projection matrices for each input variable, or interaction effect, of interest. This process is computationally expensive with an arithmetic complexity of $\mathcal{O}\left(N^3\right)$, where N represents the number of observations. Therefore, using NObSP for test data, i.e., out-of-sample extension, is computationally expensive.

Within the context of this study, the primary contributions of this paper are twofold. On one hand, the previous work (Pena-Campos et al., 2023) is extended from static systems to dynamic systems. In this frame, while SVM has been previously employed for system identification purposes, here the methodology adapts the use of NObSP to decompose the output of the identified model of a nonlinear dynamical system into additive components. Where each additive component represents the non-linear dynamic partial contribution of each input variable to the output. These components can be further used in a framework of block system identification to recognize the system components. To the best of the authors' knowledge, the proposed algorithm is the only method capable of retrieving the partial nonlinear dynamic contribution of each independent input without specifying a priori the most relevant input variables or interaction effects of interest and without adjusting the model based on prior domain knowledge. On the other hand, this methodology adapts and validates a computationally more efficient out-of-sample extension, previously introduced by the authors for static regression models (Pena-Campos et al., 2023), for the specific context of nonlinear dynamic system identification. This extension decomposes the output of the model for new testing data using only kernel evaluations and matrix multiplication between the kernel matrix and a set of coefficients. Calculating the matrix multiplication, the extension reduces the arithmetic complexity of the algorithm from $\mathcal{O}\left(N^3\right)$ to $\mathcal{O}\left(Nd^2\right)$, being d the number of support vectors.

This paper is organized as follows: in Section 2, the extension of NObSP for dynamic models using SVM, as well as the out-of-sample extension, are presented. Here, an example is used to evaluate the performance of NObSP. Section 3 presents the results of NObSP using the toy dataset. Section 4 discusses the results and possible improvements to the algorithm. Finally, Section 5 presents some conclusions from the results obtained using the proposed method.

2 Methods

The analysis begins by considering the standard mathematical structure used to represent a broad class of discrete-time, multi-input, single-output (MISO) nonlinear dynamic systems (Ljung et al., 2020). This model, often referred to as a Nonlinear Moving Average (NMA) or Nonlinear Finite Impulse Response (NFIR) model, represents the system output as a general, unknown function of a regressor vector composed of present and past inputs.

This generalist structure is foundational in system identification as it can approximate a wide variety of nonlinear dynamics, including, but not limited to, block-oriented structures such as the Wiener and Hammerstein models discussed in this work. The model is defined as:

$$\begin{aligned} y[n] \;=\; & f\big(x_1[n],\ldots,x_1[n-m_1];\ldots;x_l[n],\ldots,x_l[n-m_l];\ldots; \\ & x_r[n],\ldots,x_r[n-m_r]\big)+\eta, \end{aligned} \tag{1}$$

where $n \in \mathbb{Z}^+$ is the discrete-time variable, $y[n] \in \mathbb{R}$ is the n^{th} sample of the model's output, and η is a Gaussian noise. The function $f(\cdot)$ is an unknown nonlinear continuous map that depends on a regressor vector built from the history of the r input variables $x_l[n] \in \mathbb{R}$. Each of the r inputs contributes a block of its current value (e.g., $x_l[n]$) and its own m_l past delays (e.g., $x_l[n-1],\ldots,x_l[n-m_l]$). Here, r represents the number of physical input variables (e.g., voltage, flow rate), and m_l represents the system's memory (number of delays) associated with that specific l^{th} input. Therefore, the total dimension of the regressor vector is the sum of all these contributions: r (for the r current inputs) plus the sum of all individual delays, $\mathbf{M}$, where $\mathbf{M}=\sum_{l=1}^{r} m_l$. The total dimension is thus $r+\mathbf{M}$. The Gaussian noise term η is included to represent unmodeled dynamics or measurement noise, thus creating a more realistic identification scenario. Now, let consider the decomposition of $y[n]$ as an additive sum of the form

$$\begin{aligned} y[n] = & \sum_{l=1}^{r} f_l\left(x_l[n],\ldots,x_l[n-m_l]\right)+ \\ & \sum_{l=1}^{r}\sum_{h>l}^{r} f_{l,h}\left(x_l[n],\ldots,x_l[n-m_l];\ldots;x_h[n],\ \ldots,\right. \\ & \left. x_h[n-m_h]\right)+\mathbf{G}+\eta, \end{aligned} \tag{2}$$

where $f_l:\mathbb{R}^{m_l+1} \longrightarrow \mathbb{R}$ represents the map that indicates the (non)linear contribution of the l^{th}-input variable on the output, $f_{l,h}:\mathbb{R}^{m_l+m_h+2} \longrightarrow \mathbb{R}$ represents the map that indicates the (non)linear second-order interaction contribution of the variables x_l and x_h on the output, and $\mathbf{G} \in \mathbb{R}$ represents the (non)linear contributions of higher order interactions.

The main goal of this paper is to introduce an algorithm that allows the decomposition of the function f, which represents the nonlinear dynamic system presented in Equation 1, into the sum presented in Equation 2. In this framework, the term $\mathbf{G}$ represents the blended contribution of all second- and higher-order interactions. A primary objective of the proposed methodology is to quantify the relative importance of this term. As will be detailed in Section 2.2 (see Equation 13), the magnitude of this residual term—i.e., the difference between the full model output and the sum of the first-order components—is used precisely to determine when higher-order interactions are dominant or when the system can be considered negligibly non-additive.

2.1 Support vector machines for system identification

SVM is a kernel-based methodology that can be used either for regression or classification problems. A system identification problem can be formulated as a regression problem, where the input variables contain a temporal window of the input data. In the primal space, the model has the form

$$\hat{y}[n]=\boldsymbol{\omega}^{T}\varphi(\mathbf{x}[n])+b, \tag{3}$$

where $\hat{y}[n] \in \mathbb{R}$ is the output of the model, $\mathbf{x}[n] \in \mathbb{R}^{r(m+1)}$ is the vector containing the inputs of the model and their delays, r represents the number of input variables, m is the length of the temporal window and b is the bias term. For simplicity, it is assumed that all variables have the same number of delays in the model, but this is not a restriction. In fact $\mathbf{x}[n]$ is a column vector of the form $\mathbf{x}[n]=[\mathbf{x}_1[n],\cdots,\mathbf{x}_r[n]]^T$, with $\mathbf{x}_l[n] \in \mathbb{R}^{(m+1)}$ and $\mathbf{x}_l[n]=[x_l[n],\cdots,x_l[n-m]]$ is the entry for the l^{th}-input regressor (e.g., the u_1 block in the example of the previous section) and T represents the transpose. For simplicity, this study assumes a uniform memory m for all inputs. The selection of an optimal m (or different m_l for each input) is a separate model selection problem, which falls outside the scope of interpreting the already-identified model. Besides, $\varphi : \mathbb{R}^{r\cdot(m+1)} \longrightarrow \mathbb{R}^p$ represents the nonlinear mapping of the input vector into a high-dimensional (and potentially infinite) feature space. This "kernel trick" is the core of the SVM, allowing it to solve a linear regression problem in the feature space, which corresponds to a powerful nonlinear regression in the original input space (Suykens J. A. K. et al., 2002). Moreover, $\boldsymbol{\omega} \in \mathbb{R}^p$ are the weights of the SVM model. Depending on the selection of the kernel function, the dimension p of the nonlinear mapping can be infinite.

Before the decomposition of Equation 2 can be performed, an accurate, non-parametric model of the full function f in Equation 1 must first be identified. Therefore, the process begins by determining how to obtain f using SVM.

To make this structure concrete, consider a simple MISO system with $r = 2$ inputs (u_1, u_2) and a uniform memory of $m = 2$. The general input vector $\mathbf{x}[n]$ from Equation 3 for any time n is constructed by "stacking" the present and past values of all inputs $\mathbf{x}[n] = \left[\left[u_1[n], u_1[n-1], u_1[n-2]\right]^T \Big| \left[u_2[n], u_2[n-1], u_2[n-2]\right]^T\right]$.

The SVM model will learn a single function $f(\mathbf{x}[n])$ based on this vector. The goal of NObSP is to decompose the output of this function, $\hat{y}[n]$, back into two components: one attributable to the block $[u_1[n], u_1[n-1], u_1[n-2]]^T$ and another attributable to the block $[u_2[n], u_2[n-1], u_2[n-2]]^T$.

Given the training set $\{\mathbf{x}^{(i)}, y^{(i)}\}_{i=1}^{N}$, where the superindex (i) indicates the i^{th} observation, the SVM regression problem can be formulated as follows (Suykens J. A. K. et al., 2002):

$$\begin{aligned}
\min_{\omega, b, \xi, \xi^*} \quad & \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} + c\sum_{i=1}^{N}\left(\xi^{(i)} + \xi^{*(i)}\right) \\
\text{subject to} \quad & y^{(i)} - \boldsymbol{\omega}^T\varphi(\mathbf{x}^{(i)}) - b \le \epsilon + \xi^{(i)}, \\
& \boldsymbol{\omega}^T\varphi(\mathbf{x}^{(i)}) + b - y^{(i)} \le \epsilon + \xi^{*(i)}, \\
& \xi^{(i)}, \xi^{*(i)} \ge 0, \\
& \qquad\qquad\qquad \forall i = 1, \ldots, N
\end{aligned} \tag{4}$$

where c is the regularization constrain parameter, in the Vapnik ϵ-insensitive loss function ϵ is the tolerated error for the regression model, $\xi^{(i)}$ and $\xi^{*(i)}$ are slack variables that manage data outside the ϵ-sensitive tube, and N is the number of observations. Taking the Lagrangian and solving for the Karush-Kuhn-Tucker conditions for optimally, the solution to problem Equation 4 in matrix form is given by Suykens J. A. K. et al. (2002).

$$\hat{\boldsymbol{y}} = \boldsymbol{\Omega}\boldsymbol{\alpha} + b, \tag{5}$$

where $\hat{\boldsymbol{y}} \in \mathbb{R}^N$ is a column vector representing the output of the model, with components such that $\hat{\boldsymbol{y}} = [\hat{y}^{(1)}, \cdots, \hat{y}^{(N)}]^T$, $\boldsymbol{\Omega} \in \mathbb{R}^{N\times d}$, with d the number of support vectors, $\boldsymbol{\Omega}^{(i,j)} = \varphi(\mathbf{x}^{(i)})^T\varphi(\mathbf{x}^{(j)}) = \mathrm{K}(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$ is the ij^{th} element of the kernel matrix, and $\mathrm{K}(\cdot,\cdot)$ is the kernel function, considering $x^{(i)}$ and $x^{(j)}$ the i-th and j-th observation, respectively. Besides, $\boldsymbol{\alpha} \in \mathbb{R}^d$ is a column vector containing the Lagrange multipliers and b is the bias term. To solve the problem (Equation 4), it is necessary to find the values for the kernel hyper-parameters, as well as the model order m, that minimizes the cost function.

Furthermore, for NObSP, it is crucial to normalize the input data and center the kernel matrix. This normalization can be done by subtracting the bias term in the estimated output such that (Suykens J. et al., 2002).

$$\hat{\boldsymbol{y}} = \boldsymbol{\Omega}_C\boldsymbol{\alpha}, \tag{6}$$

where $\boldsymbol{\Omega}_C = \mathbf{M}_1\boldsymbol{\Omega}\mathbf{M}_2$ is the centered kernel matrix, $\mathbf{M}_1 = \mathbf{I}_N - \mathbf{1}_N\mathbf{1}_N^T/N$ and $\mathbf{M}_2 = \mathbf{I}_d - \mathbf{1}_d\mathbf{1}_d^T/d$ are centering matrices, with $\mathbf{I}_N$ the identity matrix of N size, and $\mathbf{1}_N \in \mathbb{R}^N$ is a column vector of unitary entries. The ij^{th} entry of the centered kernel matrix is given by $\boldsymbol{\Omega}_C^{(i,j)} = (\varphi(\mathbf{x}^{(i)}) - \mu_\varphi)^T(\varphi(\mathbf{x}^{(j)}) - \mu_\varphi)$, with $\mu_\varphi \in \mathbb{R}^p$ the mean value of the nonlinear transformation of the input variables.

Since SVM is used to identify the nonlinear model, then the SVM model is able to represent, in a non-parametric way (Equation 1). Then, the function $f(\cdot)$ should be a function that lies in a Hilbert space, i.e., $f(\cdot)$ is a smooth and continuous function.

It is critical to understand the relationship between the SVM model (Equation 3) and the decomposition goal (Equation 2). The SVM does not inherently support the decomposition. Rather, the trained SVM model becomes the black-box function f that is to be interpreted. The SVM provides a non-parametric representation of the overall system dynamics, $\hat{y}[n] = f(\mathbf{x}[n])$.

The NObSP methodology, introduced next, is the *post-hoc* tool that operates on this already-trained SVM model. NObSP takes the full model f and applies a geometric decomposition to retrieve the non-parametric partial contributions $f_l(\cdot)$, $f_{l,h}(\cdot)$, etc., which aligns with the additive structure defined in Equation 2.

2.2 Nonlinear oblique subspace projections for SVM

The core challenge in functional decomposition, and the primary motivation for NObSP, arises when input regressors are correlated. This collinearity in the input space is the underlying cause for the subspaces in the nonlinear feature space (the high-dimensional space mapped by $\varphi(\cdot)$) to be non-orthogonal. In this feature space, the SVM model itself is linear (Equation 5). However, the input correlation means that their respective feature subspaces are not orthogonal; their intersection is not null and they are overlapping.

A standard orthogonal projection (which assumes orthogonality) would incorrectly capture energy from all other overlapping subspaces, leading to an erroneous decomposition. This is precisely the problem Nonlinear Oblique Subspace Projections (NObSP) is designed to address. NObSP operates geometrically in this feature space, using oblique projections to mitigate the cross-contributions from one overlapping subspace to another and isolate only the unique contribution of each regressor

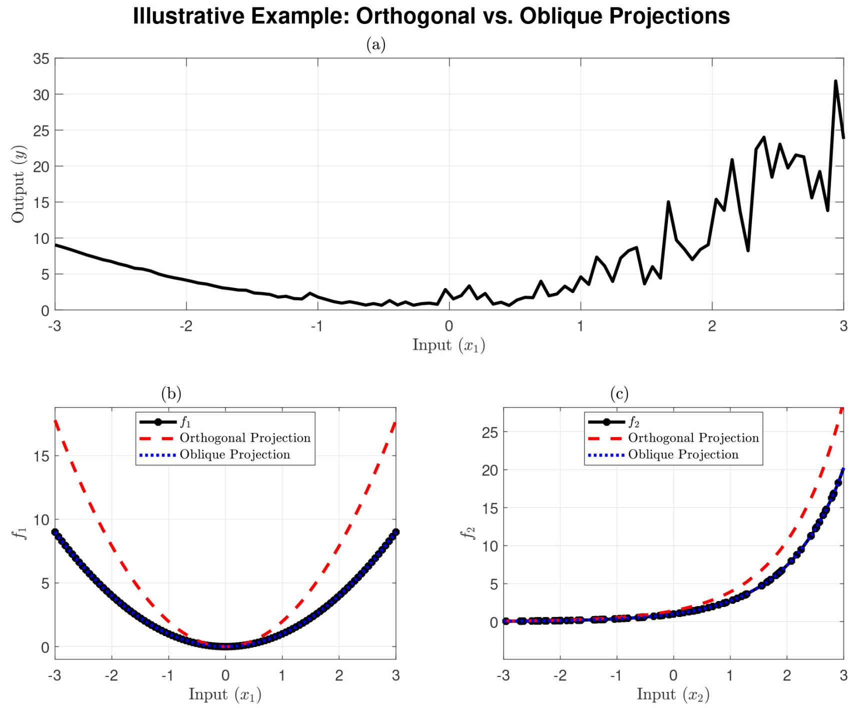

FIGURE 1
Illustrative example of NObSP for correlated regressors. **(a)** Shows the total observed signal y. **(b)** Shows the retrieval of $f_1(x_1) = x_1^2$. **(c)** Shows the retrieval of $f_2(x_2) = \exp(x_2)$. In both **(b, c)**, the orthogonal projection (red dashed line) fails to match the ground truth (black line), while the oblique projection (NObSP, blue dotted line) successfully retrieves the true component.

(Caicedo et al., 2019; Bring, 1996). It is important to note, however, that in the presence of extremely high correlations, the method can become numerically unstable and may produce unreliable results.

To illustrate this principle visually, a simple simulation was performed. Consider a MISO system $y = f_1(x_1) + f_2(x_2)$, where the true components are $f_1(x_1) = x_1^2$ and $f_2(x_2) = \exp(x_2)$. The inputs were generated such that x_2 is a noisy, correlated version of x_1 (e.g., $x_2 \approx x_1 + c$). The results are shown in Figure 1. It is important to note that while the data was generated with a dependency, the NObSP method treats x_1 and x_2 as independent regressors. When the algorithm isolates the f_1 contribution (conceptually “zeroing” x_2), it is treating x_2 as an independent subspace, not as a constant c.

As predicted by the theory, Figures 1b, c show that the orthogonal projection fails, retrieving a blended, incorrect signal. In contrast, the NObSP (oblique) projection successfully isolates the unique contribution of both $f_1(x_1)$ and $f_2(x_2)$, matching the ground truth. This simple example demonstrates the principle that NObSP extends to the high-dimensional feature space of the SVM. The following defines this nonlinear extension formally.

Let define $\mathbf{A} = [\mathbf{A}_l \quad \mathbf{A}_{(l)}]$ as a matrix where its columns span the subspace $\mathcal{V} \subset \mathbb{R}^N$, with $\mathbf{A} \in \mathbb{R}^{N\times r}$, $\mathbf{A}_l \in \mathbb{R}^{N\times q}$ a partition of $\mathbf{A}$ that spans the subspace $\mathcal{V}_l \subset \mathcal{V}$ and $\mathbf{A}_{(l)} \in \mathbb{R}^{N\times(r-q)}$ a partition of $\mathbf{A}$ that spans the subspace $\mathcal{V}_{(l)} \subset \mathcal{V}$. Considering $\mathcal{V} = \mathcal{V}_1 \oplus \cdots \oplus \mathcal{V}_l \oplus \cdots \oplus \mathcal{V}_r$, with $\oplus$ the direct sum operator and r the number of regressor subspaces embedded in $\mathbf{A}$, then the matrix that represents the oblique subspace projection onto $\mathcal{V}_l$ along $\mathcal{V}_{(l)} = \mathcal{V}_1 \oplus \cdots \oplus \mathcal{V}_{l-1} \oplus \mathcal{V}_{l+1} \oplus \cdots \oplus \mathcal{V}_r$, i.e., the matrix that projects onto the subspace of the input regressor $\mathbf{x}_l$ along the complementary regressors subspace $\mathbf{x}_{(l)}$, is denoted by $\mathbf{P}_{l/(l)}$ and it is defined as

$$\mathbf{P}_{l/(l)} = \mathbf{A}_l(\mathbf{A}_l^{\mathrm{T}}\mathbf{Q}_{(l)}\mathbf{A}_l)^{\dagger}\mathbf{A}_l^{\mathrm{T}}\mathbf{Q}_{(l)}, \tag{7}$$

where $\dagger$ is the generalized inverse of Moore-Penrose (pseudoinverse), $\mathbf{Q}_{(l)}$ is the orthogonal projector onto $\mathrm{Null}(\mathbf{A}_{(l)}^{\mathrm{T}}) \subset \mathcal{V}_{(l)}^{\perp}$, computed as $\mathbf{Q}_{(l)} = \mathbf{I}_N - \mathbf{P}_{(l)}$, $\mathbf{P}_{(l)}$ is the orthogonal projector needed to find the base onto $\mathcal{V}_{(l)}$, and computed as $\mathbf{P}_{(l)} = \mathbf{A}_{(l)}(\mathbf{A}_{(l)}^{\mathrm{T}}\mathbf{A}_{(l)})^{\dagger}\mathbf{A}_{(l)}^{\mathrm{T}}$, being Null $(\cdot)$ the null space of a matrix.

Considering $\hat{y}_l = f_l(\mathbf{x}_l)$, as presented in Equation 2, the objective is to find an oblique projection matrix such that

$$\hat{\boldsymbol{y}}_l = \mathbf{P}_{l/(l)}\hat{\boldsymbol{y}}, \tag{8}$$

where $\hat{\boldsymbol{y}}$ is the output of the SVM regression model and $\mathbf{P}_{l/(l)}$ is the oblique projection matrix onto the subspace spanned by the nonlinear transformation of the l^{th} input variable, along the direction defined by the other variables, represented by (l). As presented in Caicedo et al. (2019), proper kernel evaluations can be used to obtain these projection matrices. Therefore, a base for the subspace that represents the nonlinear transformation of the input regressor $\mathbf{x}_l$ can be found by using the kernel matrix $\mathbf{\Omega}_l$, where

$$\mathbf{\Omega}_l^{(i,j)} = \mathrm{K}(\mathbf{x}_l^{(i)}, \mathbf{x}^{(j)}), \tag{9}$$

with $\mathbf{x}_l^{(i)} = [0, \cdots, x_l[i], \cdots, x_l[i-m], \cdots, 0]$. In the same way, a basis for the subspace that represents the nonlinear transformation of the complementary regressors $\mathbf{x}_{(l)}$ is defined by the kernel matrix $\mathbf{\Omega}_{(l)}$, where

$$\mathbf{\Omega}_{(l)}^{(i,j)} = \mathrm{K}(\mathbf{x}_{(l)}^{(i)}, \mathbf{x}^{(j)}), \tag{10}$$

and $\mathbf{x}_{(l)}^{(i)} = [x_1[i], \cdots, x_1[i-m], \cdots, x_{l-1}[i], \cdots, x_{l-1}[i-m], 0, \cdots, 0, x_{l+1}[i], \cdots, x_{l+1}[i-m], \cdots, x_r[i], \cdots, x_r[i-m]]$. In summary, the column space of $\mathbf{\Omega}_l$ represents the subspace for the nonlinear transformation of the variable $\mathbf{x}_l$, onto which it will project the estimated output of the model, and the column space of $\mathbf{\Omega}_{(l)}$ represents the reference subspace for the projection. Using both kernel matrices, the nonlinear oblique projection can be defined as

$$\mathbf{P}_{l/(l)} = \mathbf{\Omega}_l(\mathbf{\Omega}_l^{\mathrm{T}}\mathbf{Q}_{(l)}\mathbf{\Omega}_l)^{\dagger}\mathbf{\Omega}_l^{\mathrm{T}}\mathbf{Q}_{(l)}, \tag{11}$$

where $\mathbf{Q}_{(l)} = \mathbf{I}_N - \mathbf{P}_{(l)}$, and $\mathbf{P}_{(l)} = \mathbf{\Omega}_{(l)}(\mathbf{\Omega}_{(l)}^{\mathrm{T}}\mathbf{\Omega}_{(l)})^{\dagger}\mathbf{\Omega}_{(l)}^{\mathrm{T}}$. For more details on the proof of Equation 11, please refer to Caicedo et al. (2019).

The nonlinear version for the oblique projections presented in Equation 11 can be used to decompose the output of a dynamic model into additive components, each representing the nonlinear dynamic contribution of the input variables on the output. In this study, the focus is placed on retrieving the *first-order main effects* (i.e., $\hat{\boldsymbol{y}}_l$), as this main effect provides the most direct interpretation of each input regressor's partial contribution. It is crucial to discuss the role of second- and higher-order terms in this framework.

While the NObSP methodology can be formally extended to compute interaction effects (e.g., $\hat{\boldsymbol{y}}_{l,h}$) by defining a target subspace $\mathbf{\Omega}_{l,h}$ (Caicedo et al., 2019), this interaction effect introduces a significant interpretive challenge. The fundamental issue lies not in the projection method, but in the functional nature of the (unknown) interaction itself.

NObSP, by design, finds the total partial contribution of a regressor (e.g., $\hat{\boldsymbol{y}}_l$). If the underlying model contains a simple multiplicative interaction (e.g., $f_{l,h} = x_1 \cdot x_2$), the projection—which is analogous to setting other inputs to zero—successfully isolates the main effects, as the interaction term $\mathbf{x}_1 \cdot 0$, vanishes.

However, this behavior cannot be guaranteed for an arbitrary nonlinear function. Consider, for example, a model with a non-separable additive interaction, such as $f(x_1, x_2) = f_l(x_1) + f_h(x_2) + \cos(x_1 + x_2)$. In this case, the NObSP projection for x_1 (analogous to setting $x_2 = 0$) will correctly retrieve the total contribution $\hat{\boldsymbol{y}}_l \approx f_l(x_1) + \cos(x_1)$, and the projection for x_2 will retrieve $\hat{\boldsymbol{y}}_h \approx f_h(x_2) + \cos(x_2)$. The interaction term $\cos(\cdot)$ becomes additively coupled (or blended) with both main effects.

Thus, NObSP is performing correctly; it reveals exactly what the black-box model is doing. The challenge, which remains an open research problem, is one of decoupling: how to further separate the "pure" main effect (e.g., $f_l(x_1)$) from its share of the non-separable interaction (e.g., $\cos(x_1)$) when a pre-defined structure cannot be imposed.

Given this challenge, this paper adopts a pragmatic and clear quantitative approach. First, the sum of all identified main effects is computed as

$$\hat{\boldsymbol{y}}_{\text{main}} = \sum_{l=1}^{r} \hat{\boldsymbol{y}}_l. \tag{12}$$

Then, the interaction residual, $\mathbf{r}_{\text{interaction}}$, is defined as the difference between the full black-box model output and the sum of the main effects, i.e.,

$$\mathbf{r}_{\text{interaction}} = \hat{\boldsymbol{y}} - \hat{\boldsymbol{y}}_{\text{main}} = \hat{\boldsymbol{y}} - \sum_{l=1}^{r} \hat{\boldsymbol{y}}_l. \tag{13}$$

This residual $\mathbf{r}_{\text{interaction}}$ serves as a direct, quantitative measure of the *total contribution of all second- and higher-order non-separable interactions*. The magnitude of this residual (e.g., its variance or RMSE) directly illustrates the relative importance of these higher-order contributions. A small residual indicates the first-order decomposition is a faithful representation (the system is largely additive), while a large residual indicates that complex, high-order interactions are dominant.

Additionally, to apply NObSP to new testing data, the projection matrices should be computed, which imply to find the kernel matrices $\mathbf{\Omega}_l$ and $\mathbf{\Omega}_{(l)}$. These operations are computationally expensive.

2.3 Out-of-sample extension for NObSP

The vectors $\hat{\boldsymbol{y}}_l$ can be seen as the nonlinear contributions on the output of the l-th input feature. NObSP provides a way to compute each one of the contributions for a single input instance. To better illustrate this fact, notice that oblique projections arise naturally from a weighted least-squares problem of the form

$$\hat{\boldsymbol{\alpha}}_l = \min_{\boldsymbol{\alpha}_l} \left|\left|\mathbf{Q}_{(l)}\left(\mathbf{y} - \mathbf{\Omega}_l \boldsymbol{\alpha}_l\right)\right|\right|^2, \tag{14}$$

with $\hat{\boldsymbol{y}}_l = \mathbf{\Omega}_l \hat{\boldsymbol{\alpha}}_l$. Then, given an input instance $\mathbf{x}^{(i)} \in \mathbb{R}^r$ the model predictions could be decomposed $\hat{y}^{(i)} = \sum_{l=1}^{r} \hat{y}_l^{(i)}$, where $\hat{y}_l^{(i)} = \mathbf{\Omega}_l \boldsymbol{\alpha}_l$, $\mathbf{\Omega}_l^{(i,j)} = \mathrm{K}(\mathbf{x}_l^{(i)}, \mathbf{x}^{(j)})$, and $\mathbf{x}_l^{(i)} = [0, \cdots, x_l[i], \cdots, x_l[i-m], \cdots, 0]$, where

$$\bigcup_{k=1}^{r} \mathcal{C}(\mathbf{\Omega}_{x_k}) \subseteq \mathcal{C}(\mathbf{\Omega}),$$

where $\mathcal{C}$ represents the column space. The values $\hat{y}_l^{(i)}$ indicate the nonlinear contribution of the l-th input feature. Therefore,

$$\hat{\boldsymbol{\alpha}}_l = (\mathbf{\Omega}_l^{\mathrm{T}}\mathbf{\Omega}_l)^{\dagger}\mathbf{\Omega}_l^{\mathrm{T}}\mathbf{y}_l. \tag{15}$$

Finally, considering a new input sample $\mathbf{x} = [x_1[n], \cdots, x_1[n-m], \cdots, x_r[n], \cdots, x_r[n-m]]$, the output of the model can be approximated by

```
Input: Regressor matrix with training samples $\mathbf{X} \in \mathbb{R}^{N\times(rm+r)}$, matrix of support vectors for the model
    $\mathbf{X}_{SV} \in \mathbb{R}^{N_{SV}\times(rm+r)}$, the kernel parameters for the kernel function with $N_{SV}$ the number of support
    vectors, and the $\boldsymbol{\alpha}$ coefficients that solve the SVM problem in the dual space.
Output: Matrix which contains the first-order contributions of each input regressor on the estimated output $\mathbf{Y} \in \mathbb{R}^{(N-m+1)\times r}$,
    and the coefficients for the out-of-sample extension $\hat{\boldsymbol{\alpha}}_l \in \mathbb{R}^{N_{SV}}$ for $l = \{1, \ldots, r\}$.
$\hat{\mathbf{y}} \leftarrow \boldsymbol{\Omega}_C\boldsymbol{\alpha}$ as in Equation 6.
for $l = 1 \rightarrow r$ do
   for $i = 1 \rightarrow N$ do
      for $j = 1 \rightarrow N_{SV}$ do
         $\boldsymbol{\Omega}_l^{(i,j)} \leftarrow K\left(\mathbf{X}_l^{(i)}, \mathbf{X}_{SV}^{(j)}\right)$ as in Equation 9
         $\boldsymbol{\Omega}_{(l)}^{(i,j)} \leftarrow K\left(\mathbf{X}_{(l)}^{(i)}, \mathbf{X}_{SV}^{(j)}\right)$ as in Equation 10
      end for
   end for
   $\mathbf{P}_{(l)} \leftarrow \boldsymbol{\Omega}_{(l)}(\boldsymbol{\Omega}_{(l)}^{\top}\boldsymbol{\Omega}_{(l)})^{\dagger}\boldsymbol{\Omega}_{(l)}^{\top}$
   $\mathbf{Q}_{(l)} \leftarrow \mathbf{I}_N - \mathbf{P}_{(l)}$
   $\mathbf{P}_{l/(l)} = \boldsymbol{\Omega}_l(\boldsymbol{\Omega}_l^{\top}\mathbf{Q}_{(l)}\boldsymbol{\Omega}_l)^{\dagger}\boldsymbol{\Omega}_l^{\top}\mathbf{Q}_{(l)}$ as in Equation 11
   $\hat{\mathbf{y}}_l \leftarrow \mathbf{P}_{l/(l)}\hat{\mathbf{y}}$ as in Equation 8
   $\hat{\boldsymbol{\alpha}}_l \leftarrow (\boldsymbol{\Omega}_l^{\top}\boldsymbol{\Omega}_l)^{\dagger}\boldsymbol{\Omega}_l^{\top}\hat{\mathbf{y}}_l$ as in Equation 16
end for
$\mathbf{Y} \leftarrow$ concatenate all $\mathbf{y}_l$
```

Algorithm 1. Nonlinear Subspace Projection (NObSP).

$$\hat{y}(\mathbf{x}) \approx \sum_{l=1}^{r} \mathrm{K}(\mathbf{x}_l, \mathbf{x}_{\mathrm{SV}})\boldsymbol{\alpha}_l. \tag{16}$$

In this way, for new data points $\mathbf{x}$, the output of the model is given by $\hat{\boldsymbol{y}}_l = \mathrm{K}(\mathbf{x}_l, \mathbf{x}_{\mathrm{SV}})\boldsymbol{\alpha}_l$, where $\mathbf{x}_{\mathrm{SV}}$ are the support vectors and $\mathbf{x}_l = [0, \cdots, x_l[n], \cdots, x_l[n-m], \cdots, 0]$, with n representing the time. In addition, the partial nonlinear contribution of the l^{th} regressor does not require the computation of the oblique projections.

Algorithm 1 summarizes the computation for the decomposition of the output using NObSP and for the computation of the out-of-sample extension.

2.4 Simulation study

Figure 2 graphically represents the main objective for the simulations presented in this paper. First, an SVM regression model is trained by using the set of observations $\{u_1[n], u_2[n], y[n]\}_{n=1}^{N}$, where N represents the number of samples. Then, the dynamical system is decomposed using NObSP, which produces an additive model. Each component of the decomposed model represents the nonlinear contribution of each input variable, or the interaction effects, on the output. Finding the dynamic functional relation between each input and the output allows understanding the contribution of each variable to the output. The knowledge of the input/output dynamics relation might facilitate the management of each branch and would reduce the complexity of the model or even design simpler and appropriate control systems. For the simulations presented in this paper, interaction effects among the input variables on the output are not considered.

The simulations consider Wiener and Hammerstein block-oriented design for the nonlinear system. Specifically, this paper uses the same system design presented by Castro-Garcia and Suykens (2016). In Castro-Garcia and Suykens (2016), the researchers used the q-notation for system identification, where $q^{-1}u[n] = u[n-1]$. For the first branch, as presented by Castro-Garcia and Suykens (2016), the nonlinear block is given by

$$f_1[n] = z^3[n], \tag{17}$$

where $z[n]$ is the input sequence to the nonlinear block and $G_1(q)$ is a linear discrete-time transfer function defined in the q-operator notation:

$$G_1(q) = \frac{B_1(q)}{A_1(q)},$$

where

$$B_1(q) = 0.0089q^3 - 0.0045q^2 - 0.0045q + 0.0089,$$
$$A_1(q) = q^3 - 2.5641q^2 + 2.2185q - 0.6456.$$

For the second branch, the nonlinear block is given by

$$f_2[n] = sinc\,(z[n])\, z^2[n], \tag{18}$$

where

$$sinc\,(z[n]) = \frac{\sin(z[n])}{z[n]},$$

and the linear block is given by

$$G_2(q) = \frac{B_2(q)}{A_2(q)},$$

where

$$B_2(q) = 0.0047q^3 + 0.0142q^2 + 0.0142q + 0.0047,$$
$$A_2(q) = q^3 - 2.458q^2 + 2.262q - 0.7654.$$

The output of the model is the result of two additive components, one per branch, such that

$$y[n] = x_1[n] + x_2[n] + \eta,$$

where η is a Gaussian noise, $x_1[n]$ and $x_2[n]$ are defined below for each block-structure. The output for the Wiener system is given by

$$y_W[n] = G_1(q)f_1\,(u_1[n]) + G_2(q)f_2\,(u_2[n]) + \eta,$$

where $x_i[n] = G_i(q)f_i(u_i[n])$ represents the linear transformation $G_i(q)$ applied on the sequence $f_i(u_i[n])$, while f_1 and f_2 are

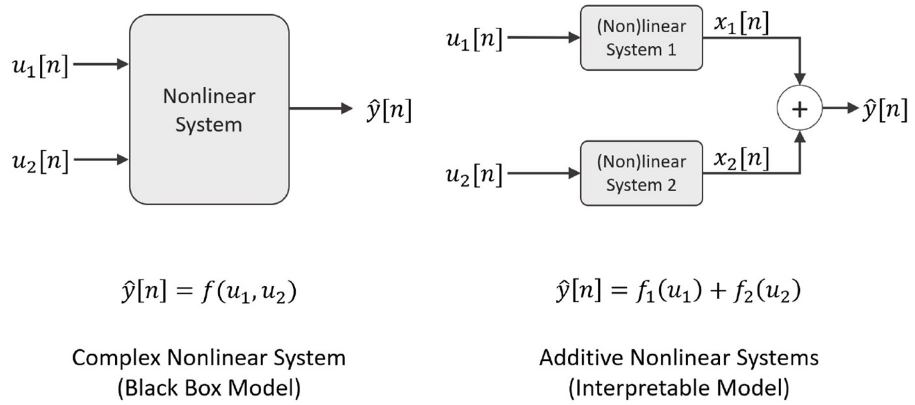
$u_1[n]$
$u_2[n]$
Nonlinear System
$\hat{y}[n]$
$u_1[n]$
(Non)linear System 1
$x_1[n]$
+
$\hat{y}[n]$
$u_2[n]$
(Non)linear System 2
$x_2[n]$
$\hat{y}[n] = f(u_1, u_2)$
Complex Nonlinear System (Black Box Model)
$\hat{y}[n] = f_1(u_1) + f_2(u_2)$
Additive Nonlinear Systems (Interpretable Model)

FIGURE 2
Proposed decomposition scheme using NObSP. The complex black-box model is decomposed into nonlinear systems that represent the decoupled nonlinear contribution of each input variable on the output.

defined in Equations 17, 18, respectively. Here, $x_i[n]$ represents the unobservable internal output of the i-th branch, which is the "ground truth" signal that the NObSP decomposition aims to retrieve. In addition, the output for a Hammerstein system is given by

$$y_H[n] = f_1\left(G_1(q)u_1[n]\right) + f_2\left(G_2(q)u_2[n]\right) + \eta,$$

where $x_i[n] = f_i\left(G_i(q)u_i[n]\right)$, for $i = \{1, 2\}$.

Two input signals were created for both scenarios to construct the training and test datasets. For $u_1[n]$, N samples were drawn from a pseudo-binary random sequence (PBRS). This signal was chosen because its broadband spectral properties are ideal for exciting a wide range of system dynamics, a standard practice in system identification. The signal was generated with the function `pbrs`, where the parameters used were an order of 99, a length of N samples, and a seed of 99 different binary elements, computed using a random sequence obtained from the function `rand`, both functions from MATLAB R2022b. For $u_2[n]$, N samples were drawn from a sinusoidal signal $u_2[n] = \sin\left(21\pi\frac{n}{N}\right)$. This signal was chosen to test the model's ability to identify and separate a purely frequency-specific component.

Since the quality of the projections depends on the performance of the model. Several simulations were performed to test the robustness of the projections.[1]

The tests performed are described below. For each simulation test (i.e., each combination of N, m, SNR, or amplitude ratio), the experiment was repeated 15 times with different random seeds for the noise generation (η). The RMSE values presented in the figures represent the average result of these 15 trials, providing a robust measure of performance.

These simulation parameters were chosen to connect with practical, real-world contexts. The number of samples N reflects the data availability from an experiment. The model order m represents the system's memory or complexity; a real-world chemical process might have a large m, while a simple electronic circuit might have a small m. Finally, the SNR reflects the quality of the measurement sensors and the level of ambient noise in a physical plant.

2.4.1 Model order and number of training samples

The impact of the model order had been evaluated m, and the number of training samples used to fit the model, N. For each pair of variable values, the Root Mean Squared Error (RMSE) was computed between the predicted output and the real output. The RMSE was also calculated for the model output and the estimated projections, i.e., the estimated nonlinear contributions of each branch $\hat{x}_1[n]$, and $\hat{x}_2[n]$.

For the simulation, the following ranges of values were used: N takes values between 50–8,000 observations, while the order of the model, m, varies between 5 and 200. Since the Wiener and the Hammerstein structures impose different dynamics on the output signal, such dynamics are expected to affect the optimal values for N and m in both block-system structures, thereby affecting the projections. The goodness of the fit was evaluated based on the estimations on a test dataset.

2.4.2 Influence of external noise

The robustness of the projections to external noise was evaluated by changing the signal-to-noise ratio (SNR) of the output signal. Simulations had been performed for values of SNR ranging from 0.8 dB up to 18 dB. This simulation used the values for N and m, in the SVM model, that produced the lowest RMSE.

The impact of changing the amplitude of noise in the output was computed for the model output, as well as for the estimated projections and the out-of-sample extension model (Equation 15).

2.4.3 Influence on the relative difference in amplitude for both branches

As shown in Figure 2, the main objective of NObSP is to decomposed the output $\hat{\mathbf{y}}$ into an additive model where $\hat{\mathbf{y}} = \mathbf{x}_1 + \mathbf{x}_2 + \mathbf{x}_{1,2}$. However, it is important to estimate the effect

[1] The code used for the simulations, for reproducibility of the results, can be found in the following repository: https://github.com/JoeCode91/EAAI_SystemDynamicIdenification_Interpretability.

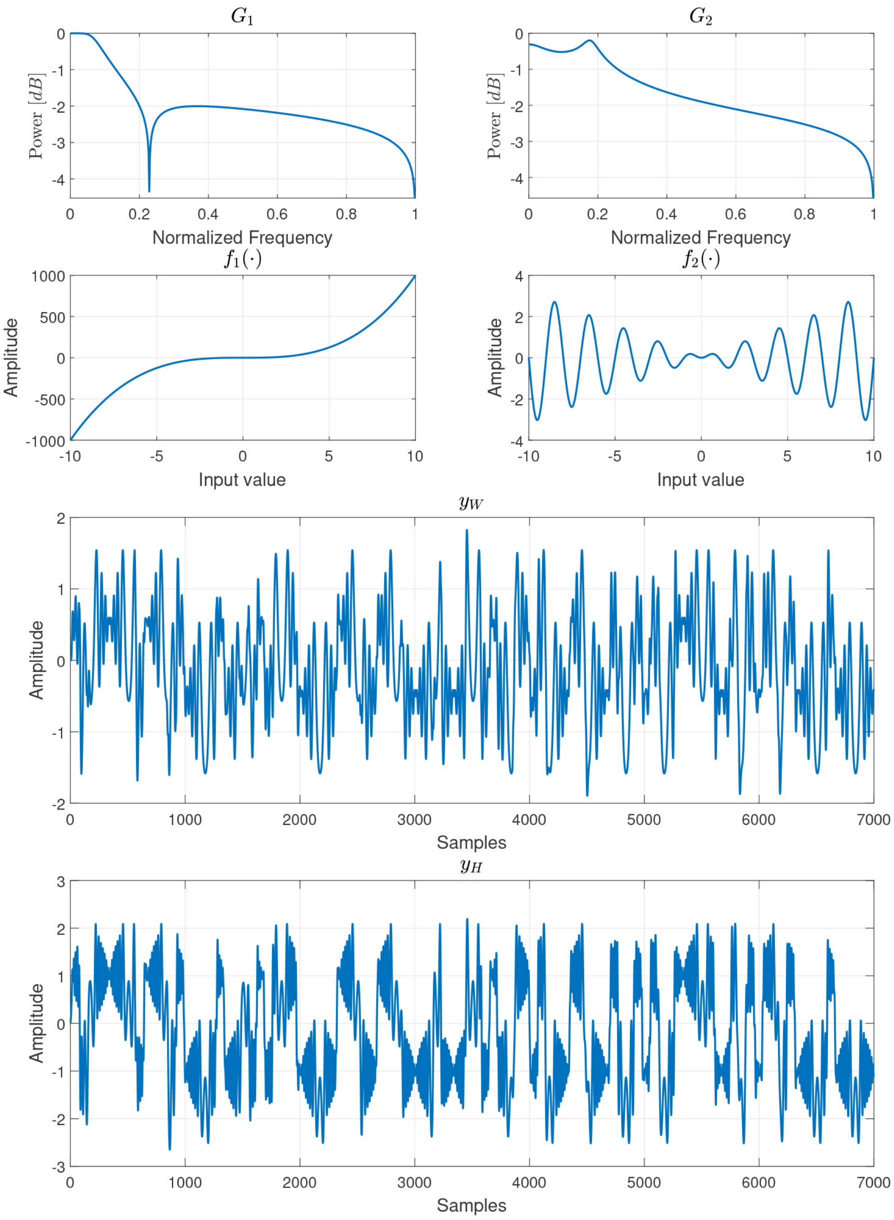

FIGURE 3
Block system approach. This figure presents the frequency response of the filters, first row, the nonlinear blocks, second row, and the output for the Wiener and the Hammerstein block-system structure, using the PBRS and the sinusoidal signal as input.

of changing the relative magnitudes of the components, i.e., what happens when the magnitude of $\mathbf{x}_1$ is larger than the magnitude of $\mathbf{x}_2$ and vice versa. In Caicedo et al. (2019), it was shown that, for the static case, NObSP is able to retrieve the dynamics of different components, even if there were differences in magnitude. The static case algorithm also determined when the contribution to the output of an input regressor was close to zero.

The impact of the changes in relative amplitude between the signal $\mathbf{x}_1$ and $\mathbf{x}_2$ is evaluated for both nonlinear block structures. The relative gain values range from 0.1 to 1, i.e., the amplitude of the signal $\mathbf{x}_1$ varies between 0.1 to 1 times the magnitude of $\mathbf{x}_2$. In total, four simulations had been performed, two for each system structure, one varying the relative amplitude of $\mathbf{x}_1$ using as reference $\mathbf{x}_2$, and varying the relative amplitude of $\mathbf{x}_2$ using $\mathbf{x}_1$ as reference.

3 Results

In Figure 3, in the top row, it is shown the normalized frequency responses for both linear systems G_1 and G_2. The second row presents the nonlinear functions used for each model branch. On the left, the function $f_1\,(n) = z^3[n]$ is presented, and on the right, the function $f_2(n) = sinc\,(z[n])\,z^2[n]$. The third row indicates the output of the Wiener system, $y_w[n]$, when using the PBRS signal as input for the first branch and the sinusoid as input for the second branch. The fourth row displays the output for the Hammerstein

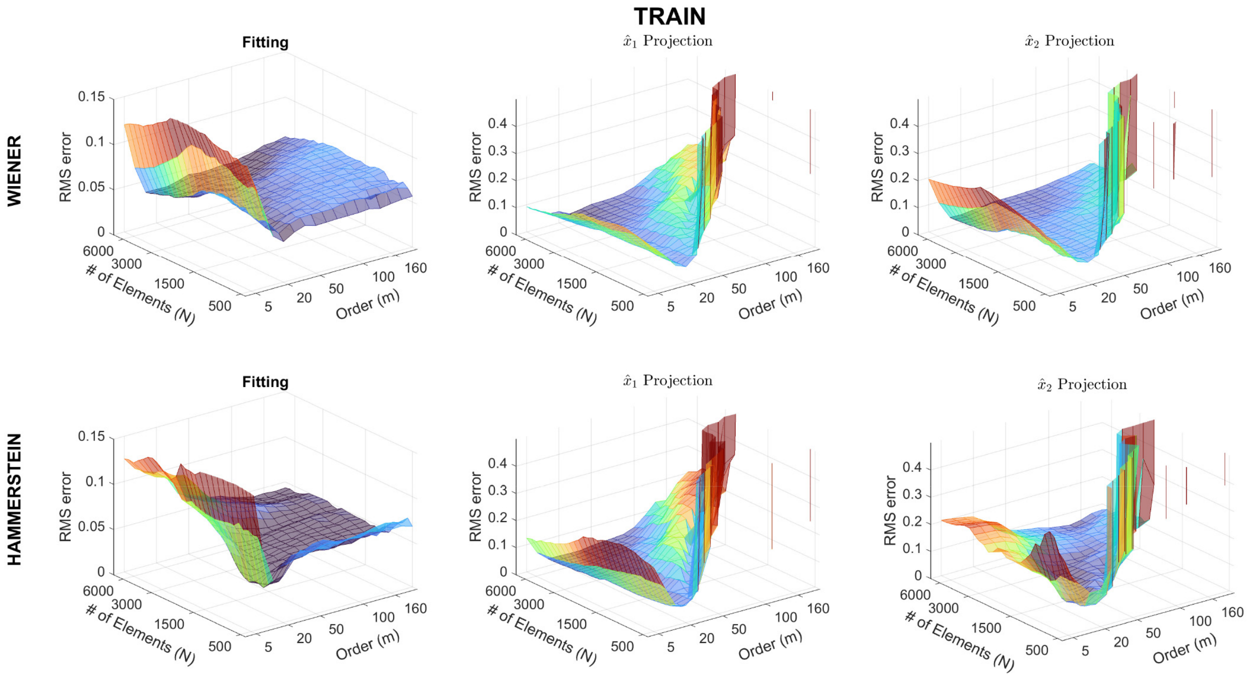

FIGURE 4
Normalized RMS Error obtained for the fit of the output, and the estimated partial (non)linear contributions of the input regressors on the output.

system, $y_H[n]$, using the same input configuration as for the Wiener case. As can be seen, the dynamics for both systems are entirely different: $y_W[n]$ presents a *smoother* behavior, mainly caused by the application of a low-pass filter after the input signals have been non-linearly transformed. While in $y_H[n]$, the nonlinear function was applied after filtering the input signals, which results in a more complex behavior than $y_W[n]$.

Figure 4 presents the evolution of the error for the different values of N and m. The first column shows the RMSE for the global fit of the model, i.e., how the model was able to fit the output using the input signals. The second and the third columns show the RMSE for the estimated projections $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$. The first row indicates the results for the Wiener structure, while the second row displays the results for the Hammerstein structure. Remarkably, for the output, first column, it can be seen that the Wiener and the Hammerstein systems present a different behavior, having a minimum for the RMSE error in different regions of the figure. For the Wiener block structure, the minimum RMSE is given by a low model order, m, but a large number of observations N. In contrast, the Hammerstein structure also requires a low model order, m, but a smaller number of observations, N. In addition, as can be seen, the RMSE for the projections behaves similarly for the Wiener and the Hammerstein structures. Besides, as the model order increases, the error in the projections increases. Notably, the minimum RMSE error in the projections is produced for the same values of N and m that minimize the RMSE for the model output, which indicates, as expected, that the performance of the decomposition algorithm is related to the performance of the model.

Figure 5 presents the results for the model fit, as well as for the estimated contributions, $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$, using test data. The first column shows the results for the Wiener structure, while the second column shows the results for the Hammerstein structure. The first row shows the estimated output of the model, the second row displays the estimated projections, and the third row presents the calibration plot to analyze the regression output. First, as can be seen in both cases, the model is able to predict the behavior of the output signal accurately. In the second row, it is shown that NObSP is able to accurately retrieve the signals $\mathbf{x}_1$ and $\mathbf{x}_2$. In the case of the Wiener structure, the dynamics of $\mathbf{x}_1$, generated using the PBRS signal, present some peaks that NObSP cannot reproduce. However, NObSP can estimate the nonlinear contributions for the general dynamics of the signal and the projections. The third row displays the calibration plots, showing that for $\hat{\mathbf{x}}_1$ in the Wiener structure, the problem with the peaks causes horizontal lines that deviate from the identity line, which represents a model without errors. The calibration plot indicates a suitable dispersion around the identity line for the other projections. However, a small span error is observed, which is caused by errors due to scaling factors.

Figure 6 presents the RMSE curves for a test set. Here, the behavior of the models has been evaluated for different values of N and m. The left columns display the results for the Wiener structure, while the right columns display the results for the Hammerstein structure. The first column shows the results using the projection matrices, and the second column shows the results from the out-of-sample extension. As expected, this figure displays a similar behavior of the RMSE for both the projections and the results obtained using the out-of-sample extension. It is important to highlight that the results for the out-of-sample extension present some peaks, which might be produced due to an ill-conditioned least-square problem, mainly caused by rank-deficient kernel matrices. This effect is more remarked for the Hammerstein system structure.

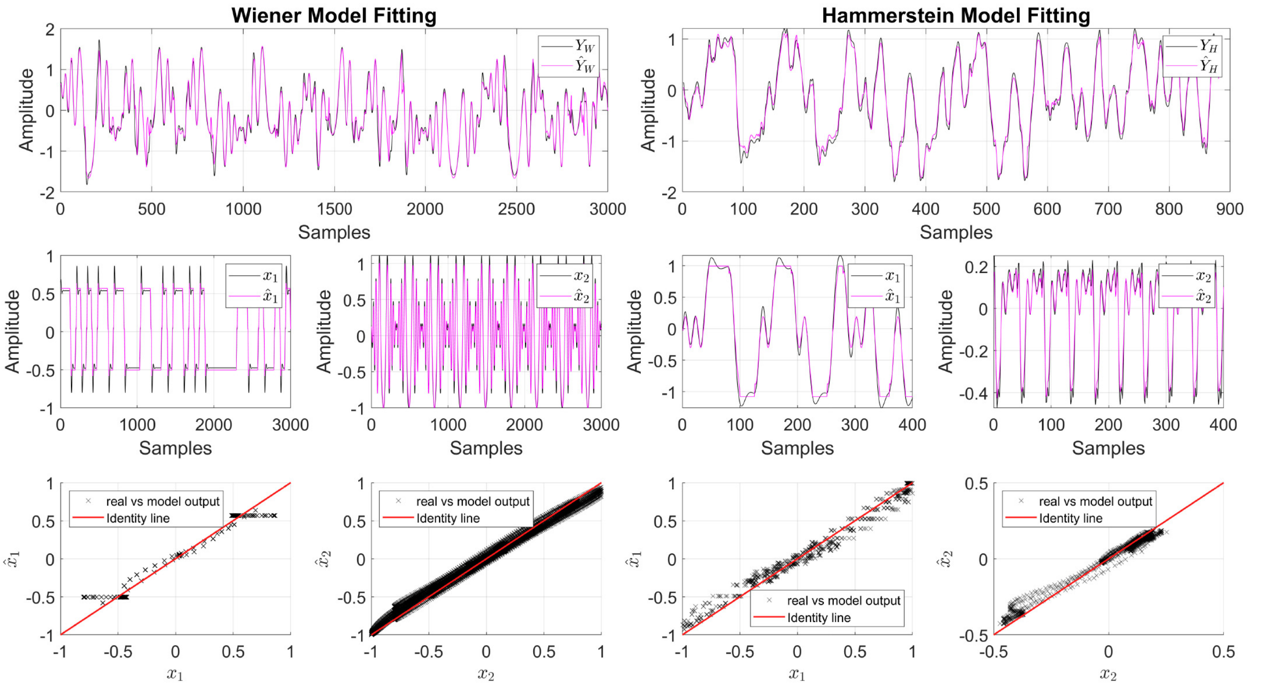

FIGURE 5
Results form NObSP applied to Wiener and Hammerstein block system structure. The first row displays the model fit in magenta and the real output in black. The second row presents the real nonlinear contribution of the input variables on the output and, in magenta, the estimated contributions using NObSP. The third row shows the calibration plots to analyze how well the algorithm predicts the projections. In some images, the x-axis limits are set for visualization effects; however, the behavior of the dynamic is similar in the entire interval.

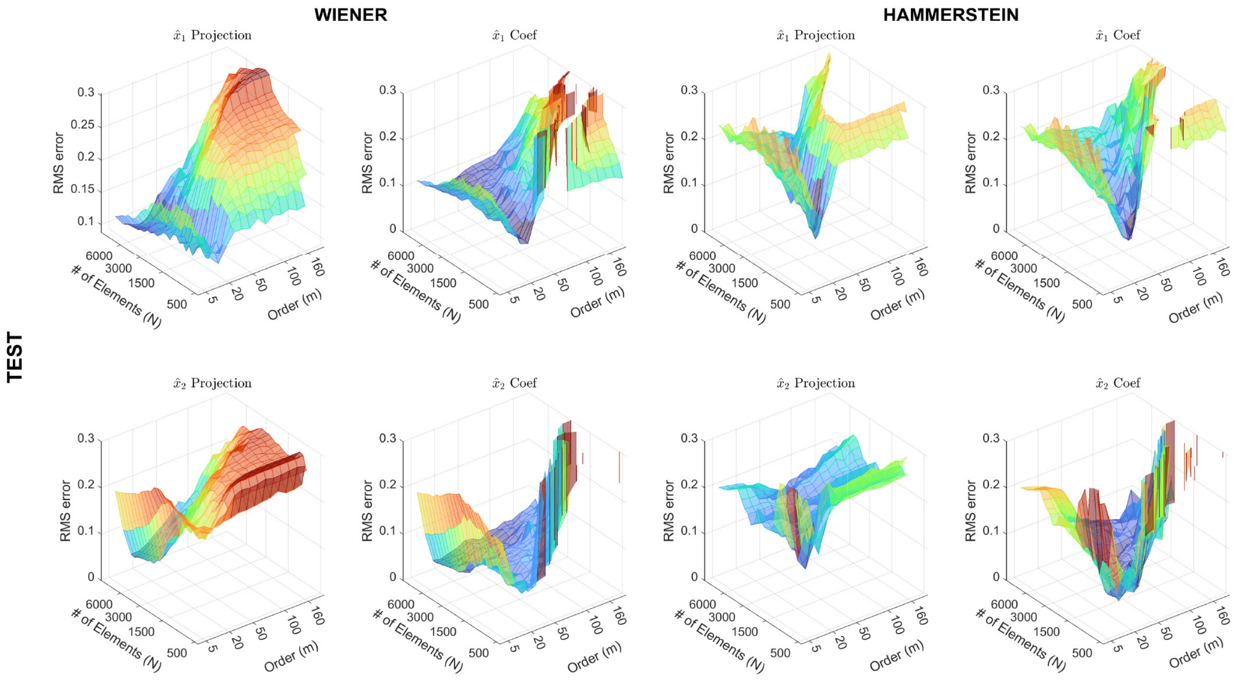

FIGURE 6
Partial (non)linear contributions of the input regressors on the output, for the test dataset, computed using the $\boldsymbol{\alpha}_l$ coefficients.

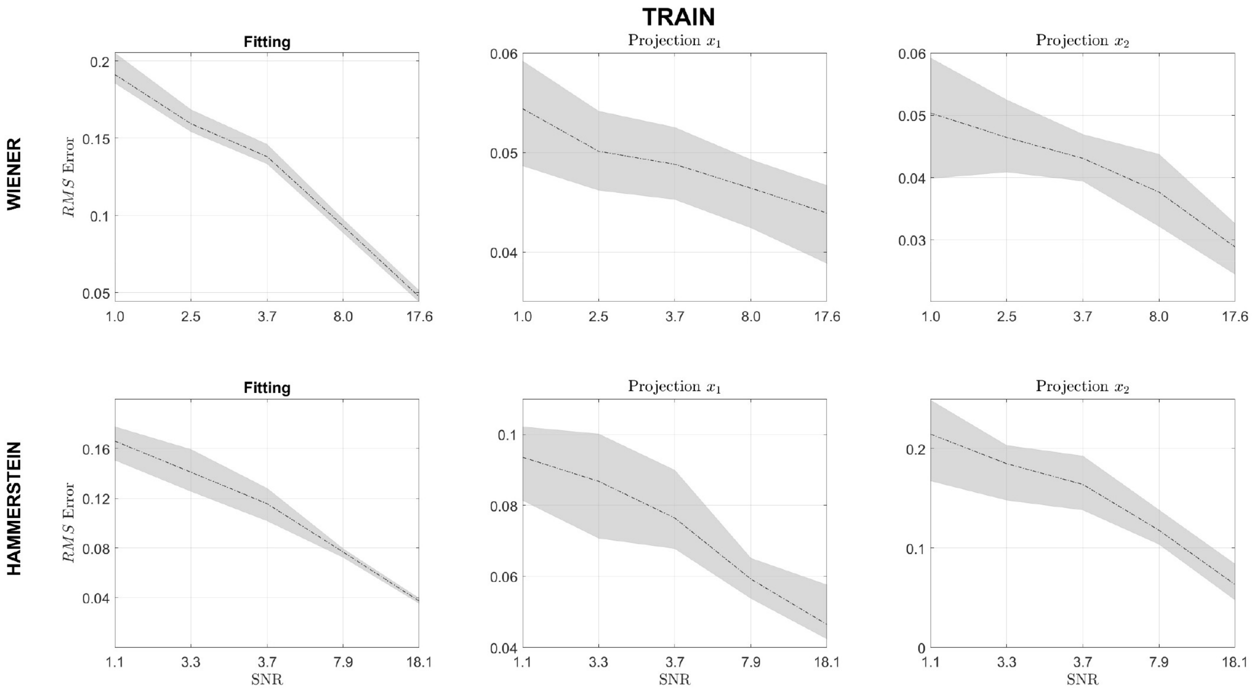

FIGURE 7
The solid black line represents the mean Normalized RMS error and the gray area the 95% confidence interval for fitting, projection $\hat{x}_1$ and projection $\hat{x}_2$ for different signal to noise ratio. The first row represents the Wiener structure, and the second one the Hammerstein structure.

Figure 7 displays the results for the changes in SNR. The first row presents the results for the Wiener block structure and the second for the Hammerstein block structure. The first column shows the results for the output fit, while the second and the third ones present the results for the projections $\hat{\boldsymbol{x}}_1$ and $\hat{\boldsymbol{x}}_2$. The figure shows that when SNR increases, the fit error decreases for both the Wiener and the Hammerstein systems, which is to be expected. Additionally, the projections, $\hat{\boldsymbol{x}}_1$ and $\hat{\boldsymbol{x}}_2$, for the Wiener system seem to be independent of the SNR within the range of variation. Similarly, the projections for the Hammerstein system seem to decrease; however, they exhibit a higher RMSE than the Wiener structure.

Figure 8 presents the results for the changes in the relative amplitude between the signals $\mathbf{x}_1$ and $\mathbf{x}_2$. As can be seen, changing the relative amplitude of $\mathbf{x}_1$ in relation to $\mathbf{x}_2$, in both system structures, increases the error in the model fit, although not significantly. However, the error of the projections decreases. When changing the relative amplitude of $\mathbf{x}_2$ in relation to $\mathbf{x}_1$. It can be seen that the fit of the model improves when the amplitudes of the signal are equivalent. As in the previous case, the projection errors also decrease.

3.1 Computational performance validation

To quantitatively validate the practical impact of the computational cost reduction, a timing analysis was conducted. The experiment compares the execution time of two methods for decomposing the model output $\hat{\boldsymbol{y}}$:

1. **NObSP:** The original approach, which requires the re-computation of the full projection matrices $\mathbf{P}_{l/(l)}$ (Equation 11) for the entire dataset. The complexity of this method is dominated by operations on the $N \times N$ kernel matrices, leading to $\mathcal{O}(N^3)$ complexity.
2. **Out-of-sample extension:** The efficient method (Equation 16), which calculates the $\hat{\boldsymbol{\alpha}}_l$ coefficients. The complexity of this step scales with $\mathcal{O}(Nd^2)$, where d is the number of support vectors and, critically, $d \ll N$.

The average execution time (over 15 simulations) was recorded for both methods across the full range of model orders (m) and sample sizes (N) used in the study. "Speedup Ratio" was then calculated as $\frac{\text{Time(Full NObSP)}}{\text{Time(OOS Extension)}}$ to quantify the relative performance gain.

Figure 9 presents the results of this analysis for both the Wiener and Hammerstein system simulations. The data clearly shows a dramatic increase in the speedup ratio as the number of samples N grows, while the model order m has a comparatively minor impact. For the Wiener system, the OOS extension was found to be over 75 times faster, and for the Hammerstein system, over 147 times faster, at the largest sample size ($N = 8,000$).

A non-monotonic behavior is also observed (i.e., the speedup ratio occasionally drops as N increases). This is likely attributable to the numerical properties of the kernel matrices being solved in NObSP method (Equation 16). The efficiency of the underlying numerical solver used to compute the pseudoinverse can vary depending on the specific characteristics of the matrix at different scales (e.g., its condition number), causing these local variations in computational time.

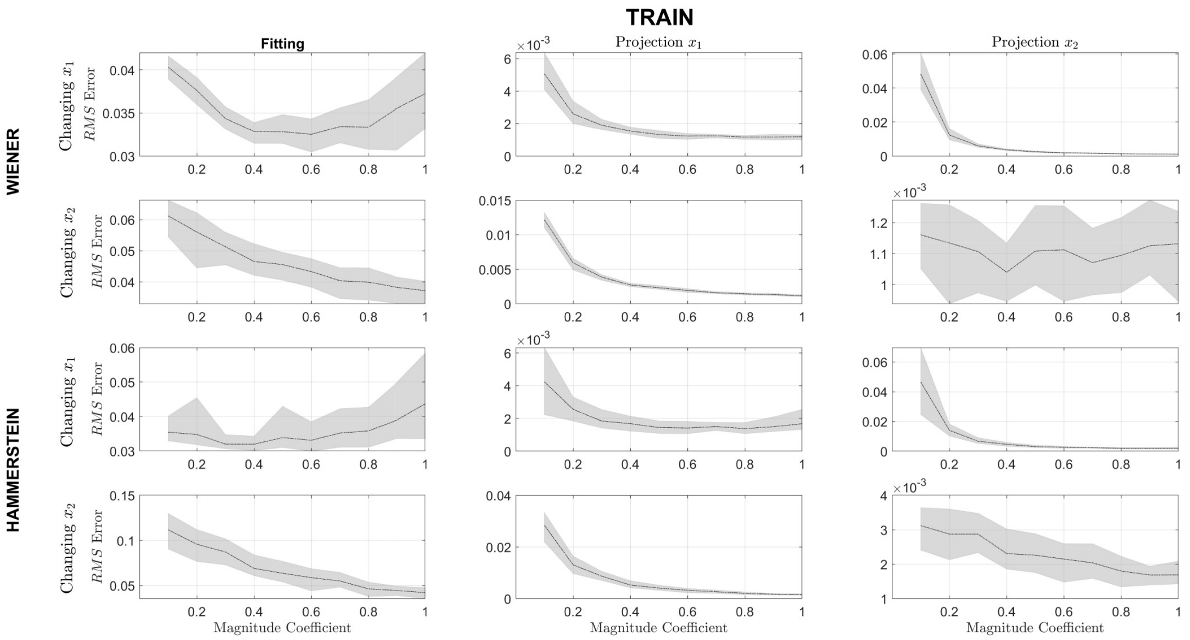

FIGURE 8
The solid black line represents the mean Normalized RMS error and the gray area the 95% confidence interval for Fitting, Projection $\hat{x}_1$ and Projection $\hat{x}_2$ for different magnitude coefficient. The upper half of the rows represent the Wiener structure, and the lower half represents the Hammerstein. In the first and third row, the changing signal is the *PRBS* signal (u_1) and in the second and fourth row, the changing signal is the *sine* signal (u_2).

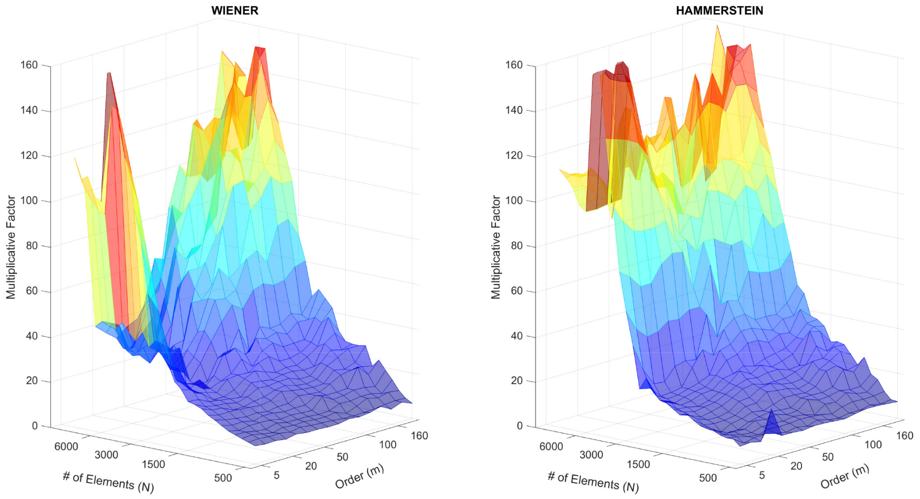

FIGURE 9
Computational speedup (ratio) of the Out-of-Sample (OOS) Extension compared to the NObSP method. The speedup, calculated as Time(NObSP)/Time(OOS Extension), is plotted as a function of the number of samples (N) and the model order (m) for both the **(left)** Wiener and **(right)** Hammerstein structures. The results demonstrate a significant performance gain that scales dramatically with N.

Despite this numerical variance, the overall behavior is consistent with the theoretical complexity. The $\mathcal{O}(N^3)$ complexity of the Full NObSP method causes its execution time to grow cubically, becoming computationally prohibitive for large datasets. Conversely, the OOS extension's cost scales much more favorably, as it avoids operations on the large $N \times N$ matrices. This experiment provides strong quantitative validation—demonstrating a speedup of more than two orders of magnitude—that the OOS extension is not just theoretically, but also practically, superior for large-scale system identification problems.

4 Discussion

This paper presents an extension of NObSP for the decomposition of the output of a nonlinear dynamical system. The extension allows the decomposition of a model, initially identified as a nonlinear mean average system using SVM, into additive components where each one represents the nonlinear contribution of each input signal on the dynamics of the output. In the literature, some methodologies exist that allow retrieving the functional relationship between inputs-outputs in a black-box model. However, these methods require to define a priori the relevant effects of interest for the designer, which can bias the identified model, e.g., functional ANOVA models (Abramovich and Angelini, 2006). Other *post hoc* methods can retrieve the functional relation for static models, such as the Partial Dependence plot (PDP) (Burkart and Huber, 2021). In addition, Volterra series can be used for dynamic system identification. However, a review of the literature suggests that this method does not allow to estimate the partial nonlinear dynamic contribution of each independent input (Cheng et al., 2017; Dalla Libera et al., 2021). Likewise, in Pei et al. (2022), the authors proposed a framework to replicate traditional methods for structural health monitoring using sigmoidal neural networks, which improves the interpretability of the resulting model. Pei et al. (2022), used domain knowledge to identify the dominant features and approximate their contributions using sigmoidal neural networks, thereby fitting the specified function approximation using a linear combination of these learned features. Even though these sigmoidal functions are generic, this method is specific for NN and imposes a specific model architecture by the association of some types of nonlinearities, which can incorporate domain knowledge. Additionally, in the context of optimization, this training method produces a local search. In contrast, NObSP is a more general framework since it allows the retrieval of the nonlinear contribution of the inputs without additional processing from an already trained black box model. In this sense, the global search developed by the minimization of the approximation error in the training process is not biased by NObSP methodology, and the functional relation between the input regressors and the output is not restricted by any condition. These characteristics allow the use of NObSP in different domains without the necessity of domain knowledge. NObSP was initially developed for LS-SVM static regression models by Caicedo et al. (2019). This study shows that NObSP is able to retrieve the partial nonlinear dynamic contribution of each input variable on the output for Wiener and Hammerstein block-system structures. In Figure 4, it is observed that the goodness of the model fit directly impacts the performance of NObSP. Therefore, in order to obtain an adequate decomposition, it is crucial first to have a model that fits the data satisfactorily.

In general, SVM performs a nonlinear transformation on the input data and maps it to a Hilbert space, facilitating the solution of the regression problem on the transformed space since, in this space, the model is considered linear. For this reason, SVM is only able to identify functions that lie on a Hilbert space. Interestingly, the kernel matrix represents a low-rank approximation of the hyperplane where the transformed data lies, which in turn means a low-rank approximation of the manifold of the identified model. Since the number of observations, N, determines the size of the kernel matrices, it impacts the projections due to the fact that the maximum rank for the kernel matrix is N. More specifically, the size of the kernel determines the number of vectors that represent the column space of the hyperplane in the Hilbert space. Considering this fact, Wiener structures seem to generate subspaces of larger dimensions, while the Hammerstein model needs fewer basis vectors to span the subspace of the nonlinear transformations. This issue might indicate that the manifold where the input and output observations lie might be more complex for the Wiener structure than for the Hammerstein structure.

Another approach to the out-of-sample extension is to consider that $\hat{\mathbf{y}}_l = \mathbf{P}_{l/(l)}\hat{\mathbf{y}}$, then $\hat{\mathbf{y}}_{\mathbf{l}} = \mathbf{P}_{l/(l)}\mathbf{\Omega}_C\boldsymbol{\alpha} = \mathbf{\Omega}_C\boldsymbol{\alpha}_l$, where $\boldsymbol{\alpha}_l \in \mathbb{R}^r$. And solving using least squares, it is obtained $\hat{\boldsymbol{\alpha}}_l = (\mathbf{\Omega}_C^{\mathrm{T}}\mathbf{\Omega}_C)^{\dagger}\mathbf{\Omega}_C^{\mathrm{T}}\hat{\mathbf{y}}_l$. The decomposition of the output model $\hat{\mathbf{y}}_{x_j} = \mathrm{K}(\mathbf{x}, \mathbf{x}_{\mathrm{SV}})\hat{\boldsymbol{\alpha}}_{x_j}$ retrieved by the $\hat{\boldsymbol{\alpha}}_{x_j}$ coefficients just need one evaluation of the kernel but the output is noisier than that given by the projections. This issue may be caused by the fact that there is an overlap of the subspaces, i.e., $\bigcap_{k=1}^{r} \mathcal{C}(\mathbf{\Omega}_k) \neq \emptyset$, where $\mathcal{C}(\mathbf{\Omega}_j) = Span(\mathbf{\Omega}_j)$. The subspaces for each input variable are not disjointed, probably due to the low-rank approximation caused by the number of support vectors and the number of data samples. Nevertheless, the decomposition of the output model using $\hat{\boldsymbol{\alpha}}_{x_j}$ coefficients obtained by solving weighted least squares, i.e., $y_{x_j} = \mathrm{K}(\mathbf{x}_j, \mathbf{x}_{\mathrm{SV}})\hat{\boldsymbol{\alpha}}_{x_j}$ produces the same results as the projection matrices since $\hat{\mathbf{y}}_{x_j} \subset \mathcal{C}(\mathbf{\Omega}_j)$. However, to compute the contribution of each input regressor, the kernel function needs to be evaluated.

In addition, concerning the out-of-sample extension, it was shown that the $\hat{\boldsymbol{\alpha}}$ coefficients can capture the dynamics of the system, largely reducing NObSP computational time and complexity. However, in some cases, ill-conditioned matrices can produce inflated coefficients, which negatively impacts the performance of the decomposition. Here, it is important to take into consideration several aspects. First, SVM is able to reproduce functions that lie in a Hilbert space. Regarding the application of the methodology to more complex systems beyond the tested Wiener and Hammerstein structures, the theoretical limits of the SVM model itself must be considered. SVMs, and specifically the LS-SVM framework used in this work, are established as universal approximators for functions that reside within a Reproducing Kernel Hilbert Space (RKHS) (Suykens J. A. K. et al., 2002; Vapnik, 1999).

These spaces are known to contain functions that are smooth and continuous. The NObSP algorithm, being a *post-hoc* geometric methodology, is designed to decompose the function *already learned* by the SVM. Therefore, it is intuited that NObSP can successfully retrieve the partial components of any system that the underlying SVM can accurately model. As long as the system's nonlinear dynamics are not so irregular as to fall outside the RKHS (e.g., highly discontinuous functions) that the SVM is capable of approximating, the decomposition is expected to be valid.

A formal mathematical proof defining the precise functional boundaries (e.g., which specific compositions of $f(\cdot)$ and $G(q)$) are identifiable by an SVM and, subsequently, decomposable by NObSP remains a complex and open question that is outside the scope of this paper. Furthermore, the practical success of this decomposition relies on the numerical stability of the kernel matrices (Equation 11), which must be well-conditioned (Caicedo et al., 2019).

Considering the changes in the SNR of the output signal, it can be seen in Figure 6 that the fit of the model and the estimated projections $\hat{x}_1$ and $\hat{x}_2$ improves at the SNR increase. This behavior is expected since the quality of the projections depends directly on the quality of the model fit. For this reason, before applying the decomposition, it is important to guarantee that the model fit is accurate. If this condition is not fulfilled, then NObSP will not perform properly. In such cases, the estimated manifold where the input and output observation lies might be under-fitted or over-fitted. The effect of each one of those phenomena on the projections is yet to be studied.

Concerning the changes in the relative magnitudes of the estimated projections, it is shown in Figure 8 that when the signals are comparable in magnitude, the normalized RMSE value is lower than when there is a significant relative difference in magnitude. Interestingly enough, increasing the magnitude of the component $\mathbf{x}_1$ reduces the RMSE for the fitting in the model output. When the PBRS signal is analyzed, it can be found that it does not belong to a Hilbert space but in a Bounded Variation space since its second derivative is discontinuous (Parhi and Nowak, 2021). Therefore, by increasing the relevance of the discontinuous components on the output, the output signal diverges more and more from a Hilbert space. In Figure 6, it can be seen that for the Wiener structure, the borders of $\hat{\boldsymbol{x}}_1$ are not well captured by NObSP, which is where the second derivative of the function is discontinuous.

To finalize, it is important to note that the main objective of this paper is not to identify whether the system contains a Wiener or a Hammerstein structure nor to propose a new identification algorithm but to provide an algorithm to decompose the output of an already identified system into independent components related to each input variable. In this sense and outside of the scope discussed in Rudin (2019), the methodology proposed in this manuscript begins with an accurate black-box model that could predict the behavior of the system. Based on this model and using a geometric approach, the interpretation algorithm projects the influence of each input regressor over the other regressors using the kernel matrix of the model, obtaining the marginal functional relation between the input regressors and the output. Once these signals are obtained, in the literature, there exist several methods that are able to identify each component of a Wiener or a Hammerstein structure (Castro-Garcia and Suykens, 2016; Bottegal et al., 2018; Bottegai et al., 2017; Falck et al., 2009). In addition, some open questions require further studies, such as: What type of composition of input signals, nonlinear functions, and linear transformations can be identified accurately using SVM? Can NObSP be applied to other nonlinear identification methods? Can NObSP allow a suitable retrieval of the second-order interaction between inputs?

5 Conclusion

This research demonstrates that Non-linear Oblique Subspace Projections (NObSP) are a viable and effective method for retrieving interpretability from black-box Support Vector Machine (SVM) models used in dynamic system identification. This work showed that a rigorous geometric decomposition, specifically one that handles correlated regressors, can successfully retrieve the blended dynamics of a nonlinear system. This moves beyond simple feature attribution by reconstructing the *full, non-parametric contribution* of each input regressor. As the simulation results in the previous section confirmed, NObSP effectively decomposed the identified SVM model into its constituent Wiener and Hammerstein sub-systems, accurately retrieving the partial dynamics. The method's robustness was also validated against significant signal noise, showing that decomposition quality improves as the Signal-to-Noise Ratio (SNR) increases.

The research findings also provide insight into the modeling of nonlinear dynamics. The analysis of the training requirements showed that the Wiener structure requires a larger number of training samples (N) than the Hammerstein structure for an accurate fit. This suggests a higher functional complexity in the identified manifold, characterized by the presence of more abrupt signal changes and discontinuities. These discontinuities require a richer, higher-dimensional basis (i.e., more support vectors) for the SVM to model accurately. On the practical side, this work validated an efficient out-of-sample extension. The computational analysis demonstrated a dramatic reduction in execution time, confirming that the $\mathcal{O}(Nd^2)$ method is scalable and practically superior in executipon time, while producing comparable decomposition results to the $\mathcal{O}(N^3)$ approach for large datasets.

Future research should focus on extending this geometric framework to other black-box models. Extending NObSP to neural networks, for example, is a significant challenge, as it would require developing a new mathematical formulation to handle the lack of an explicit kernel structure and to manage the high-dimensional, coupled nonlinearities introduced by activation functions. This validation of NObSP on sequential, dynamic data also suggests a promising direction for future work. It demonstrates that a rigorous geometric decomposition is a viable methodology for interpreting sequential data, offering a path toward true functional decomposition in other complex sequential model architectures, such as Transformers or Long Short-Term Memory (LSTMs).

Finally, while this work validated the methodology robustly using established simulation benchmarks (Wiener and Hammerstein), a crucial next step is the application of this framework to real-world experimental data. This will be essential to test the method's performance against non-ideal conditions, such as unmodeled cross-couplings and non-Gaussian noise, which are common in physical and industrial processes.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found in the article/supplementary material.

Author contributions

JP-C: Formal analysis, Funding acquisition, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. DP: Funding acquisition, Supervision, Writing – review & editing. CO-M: Funding acquisition, Supervision, Writing – review & editing. JR-F: Conceptualization, Writing – review & editing. MS-B: Conceptualization, Writing – review & editing. AC: Conceptualization, Formal analysis, Funding acquisition, Methodology, Project administration, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. This research is supported by the following funds: the Condonable Credit Doctoral Support Grant from the Academic Vice Rector's Office, the Scholarship Loan Program Grant from the Fundación para el futuro de Colombia (Colfuturo), the project PID2023-148840O B-I00 (SEAMLESS) funded by MCIN/AEI/ 10.13039/5011000 11033, the project Artificial intelligence modeling for early detection of pancreatic and stomach cancer under the ID 20346 and the project PID2023-146261OB-I00, funded by Ministry of Science and Innovation of Spain.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declare that Gen AI was used in the creation of this manuscript. To assist in editing and improving the language, grammar, and overall readability of the text.

Any alternative text (alt text) provided alongside figures in this article has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Abramovich, F., and Angelini, C. (2006). Testing in mixed-effects FANOVA models. *J. Stat. Plan. Inference* 136, 4326–4348. doi: 10.1016/j.jspi.2005.06.002

Agarwal, R., Melnick, L., Frosst, N., Zhang, X., Lengerich, B., Caruana, R., et al. (2021). "Neural additive models: interpretable machine learning with neural nets," in *Advances in Neural Information Processing Systems*, 4699–4711.

Angelini, M., Blasilli, G., Lenti, S., and Santucci, G. (2023). A visual analytics conceptual framework for explorable and steerable partial dependence analysis. *IEEE Trans. Vis. Comput. Graph*. 30, 4497–4513. doi: 10.1109/TVCG.2023.3263739

Aquino, G., Costa, M., and Filho, C. (2022). Explaining one-dimensional convolutional models in human activity recognition and biometric identification tasks. *Sensors* 22:5644. doi: 10.3390/s22155644

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., et al. (2020). Explainable Artificial Intelligence (XAI): concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. Fusion* 58, 82–115. doi: 10.1016/j.inffus.2019.12.012

Bottegai, G., Castro-Garcia, R., and Suykens, J. A. (2017). "On the identification of wiener systems with polynomial nonlinearity," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)* (IEEE), 6475–6480. doi: 10.1109/CDC.2017.8264635

Bottegal, G., Castro-Garcia, R., and Suykens, J. A. (2018). A two-experiment approach to wiener system identification. *Automatica* 93, 282–289. doi: 10.1016/j.automatica.2018.03.069

Bring, J. (1996). A geometric approach to compare variables in a regression model. *Am. Stat*. 50, 57–62. doi: 10.1080/00031305.1996.10473543

Burkart, N., and Huber, M. F. (2021). A survey on the explainability of supervised machine learning. *J. Artif. Intell. Res*. 70, 245–317. doi: 10.1613/jair.1.12228

Caicedo, A., Varon, C., Huffel, S. V., and Suykens, J. A. (2019). Functional form estimation using oblique projection matrices for ls-SVM regression models. *PLoS ONE* 14, 1–21. doi: 10.1371/journal.pone.0217967

Candon, M., Esposito, M., Fayek, H., Levinski, O., Koschel, S., Joseph, N., et al. (2022). Advanced multi-input system identification for next generation aircraft loads monitoring using linear regression, neural networks and deep learning. *Mech. Syst. Signal Process*. 171:108809. doi: 10.1016/j.ymssp.2022.108809

Castro-Garcia, R., and Suykens, J. A. (2016). "Wiener system identification using best linear approximation within the ls-svm framework," in *2016 IEEE Latin American Conference on Computational Intelligence (LA-CCI)* (IEEE), 1–6. doi: 10.1109/LA-CCI.2016.7885698

Cheng, C., Peng, Z., Zhang, W., and Meng, G. (2017). Volterra-series-based nonlinear system modeling and its engineering applications: a state-of-the-art review. *Mech. Syst. Signal Process*. 87, 340–364. doi: 10.1016/j.ymssp.2016.10.029

Dalla Libera, A., Carli, R., and Pillonetto, G. (2021). Kernel-based methods for volterra series identification. *Automatica* 129:109686. doi: 10.1016/j.automatica.2021.109686

DeVore, R., Petrova, G., and Wojtaszczyk, P. (2011). Approximation of functions of few variables in high dimensions. *Constr. Approx*. 33, 125–143. doi: 10.1007/s00365-010-9105-8

Espinoza, M., Suykens, J. A., and De Moor, B. (2005). Kernel based partially linear models and nonlinear identification. *IEEE Trans. Automat. Contr*. 50, 1602–1606. doi: 10.1109/TAC.2005.856656

Falck, T., Pelckmans, K., Suykens, J. A., and De Moor, B. (2009). Identification of wiener-hammerstein systems using ls-svms. *IFAC Proc*. 42, 820–825. doi: 10.3182/20090706-3-FR-2004.00136

Forgione, M., Muni, A., Piga, D., and Gallieri, M. (2023). On the adaptation of recurrent neural networks for system identification. *Automatica* 155:111092. doi: 10.1016/j.automatica.2023.111092

Goethals, I., Pelckmans, K., Hoegaerts, L., Suykens, J., and De Moor, B. (2005). "Subspace intersection identification of hammerstein-wiener systems," in *Proceedings of the 44th IEEE Conference on Decision and Control* (IEEE), 7108–7113. doi: 10.1109/CDC.2005.1583307

Gonzalez-Olvera, M. A., and Tang, Y. (2010). Black-box identification of a class of nonlinear systems by a recurrent neurofuzzy network. *IEEE Trans. Neural Netw*. 21, 672–679. doi: 10.1109/TNN.2010.2041068

Harrell, F. E., et al. (2001). *Regression Modeling Strategies: With Applications to Linear Models, Logistic Regression, and Survival Analysis*. Cham: Springer. doi: 10.1007/978-1-4757-3462-1

Jutte, A., Ahmed, F., Linssen, J., and Keulen, M. V. (2025). C-shap for time series: an approach to high-level temporal explanations. *ArXiv, abs/2504.11159*.

Li, F., Liang, M., He, N., and Cao, Q. (2023a). Separation identification approach for the Hammerstein-Wiener nonlinear systems with process noise using correlation analysis. *Int. J. Robust Nonl. Control* 33, 8105–8123. doi: 10.1002/rnc.6731

Li, F., Liang, M., and Luo, Y. (2023b). Correlation analysis-based parameter learning of Hammerstein nonlinear systems with output noise. *Eur. J. Control* 72:100819. doi: 10.1016/j.ejcon.2023.100819

Li, F., Qian, S., He, N., and Li, B. (2024). Estimation of wiener nonlinear systems with measurement noises utilizing correlation analysis and Kalman filter. *Int. J. Robust Nonl. Control* 34, 4706–4718. doi: 10.1002/rnc.7224

Li, F., Zheng, T., He, N., and Cao, Q. (2022). Data-driven hybrid neural fuzzy network and ARX modeling approach to practical industrial process identification. *IEEE/CAA J. Autom. Sinica* 9, 1702–1705. doi: 10.1109/JAS.2022.105821

Li, F., Zhu, X., and Cao, Q. (2023c). Parameter learning for the nonlinear system described by a class of Hammerstein models. *Circ. Syst. Signal Proc*. 42, 2635–2653. doi: 10.1007/s00034-022-02240-y

Li, J., Zong, T., and Lu, G. (2022). Parameter identification of Hammerstein–Wiener nonlinear systems with unknown time delay based on the linear variable weight particle swarm optimization. *ISA Trans*. 120, 89–98. doi: 10.1016/j.isatra.2021.03.021

Ljung, L., Andersson, C., Tiels, K., and Schön, T. B. (2020). Deep learning and system identification. *IFAC-PapersOnLine* 53, 1175–1181. doi: 10.1016/j.ifacol.2020.12.1329

Luckey, D., Fritz, H., Legatiuk, D., Peralta Abadía, J. J., Walther, C., and Smarsly, K. (2022). *Explainable Artificial Intelligence to Advance Structural Health Monitoring*. Cham: Springer International Publishing, 331–346. doi: 10.1007/978-3-030-81716-9_16

Lundberg, S. M., and Lee, S.-I. (2017). "A unified approach to interpreting model predictions," in *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17* (Red Hook, NY, USA: Curran Associates Inc.), 4768–4777.

Parhi, R., and Nowak, R. D. (2021). Banach space representer theorems for neural networks and ridge splines. *J. Mach. Learn. Res.* 22, 1960–1999. doi: 10.48550/arXiv.2006.05626

Pei, J.-S., Hougen, D. F., Kanneganti, S. T., Wright, J. P., Mai, E. C., Smyth, A. W., et al. (2022). *Interpretable Machine Learning for Function Approximation in Structural Health Monitoring*. Cham: Springer International Publishing, 369–388. doi: 10.1007/978-3-030-81716-9_18

Pena-Campos, J., Patino, D., Ocampo-Martinez, C., and Caicedo, A. (2023). "Out-of-sample extension of kernel-based interpretation models for SVM regression using oblique subspace projections," in *World Congress of the International Federation of Automatic Control. Electronic Proceedings of the IFAC World Congress 2023* (Yokohama, Japan).

Ravikumar, P., Lafferty, J., Liu, H., and Wasserman, L. (2009). Sparse additive models. *J. R. Statist. Soc. Series B* 71, 1009–1030. doi: 10.1111/j.1467-9868.2009.00718.x

Resendiz-Trejo, J. A., Yu, W., and Li, X. (2006). "Support vector machine for nonlinear system on-line identification," in *2006 3rd International Conference on Electrical and Electronics Engineering* (IEEE), 1–4. doi: 10.1109/ICEEE.2006.251894

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). "Why should i trust you?": explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16* (New York, NY, USA: ACM), 1135–1144. doi: 10.1145/2939672.2939778

Rojat, T., Puget, R., Filliat, D., Ser, J. D., Gelin, R., and Díaz-Rodríguez, N. (2021). Explainable artificial intelligence (xAI) on timeseries data: a survey. *arXiv preprint arXiv:2104.00950.*

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat. Mach. Intell.* 1, 206–215. doi: 10.1038/s42256-019-0048-x

Selvaraju, R. R., Das, A., Vedantam, R., Cogswell, M., Parikh, D., and Batra, D. (2016). Grad-cam: visual explanations from deep networks via gradient-based localization. *Int. J. Comput. Vis.* 128, 336–359. doi: 10.1007/s11263-019-01228-7

Sen, D., Deora, B. S., and Vaishnav, A. (2025). Explainable deep learning for time series analysis: integrating SHAP and LIME in LSTM-based models. *J. Inf. Syst. Eng. Manag.* 10, 412–423. doi: 10.52783/jisem.v10i16s.2627

Shi, H., Yang, N., Yang, X., and Tang, H. (2023). Clarifying relationship between pm2.5 concentrations and spatiotemporal predictors using multi-way partial dependence plots. *Remote. Sens.* 15:358. doi: 10.3390/rs15020358

Suykens, J., De Brabanter, J., Lukas, L., and Vandewalle, J. (2002). Weighted least squares support vector machines: robustness and sparse approximation. *Neurocomputing* 48, 85–105. doi: 10.1016/S0925-2312(01)00644-0

Suykens, J. A. K., Van Gestel, T., De Brabanter, J., De Moor, B., and Vandewalle, J. (2002). *Least Squares Support Vector Machines*. Singapore: World Scientific. doi: 10.1142/5089

Theissler, A., Spinnato, F., Schlegel, U., and Guidotti, R. (2022). Explainable ai for time series classification: a review, taxonomy and research directions. *IEEE Access* 10, 100700–100724. doi: 10.1109/ACCESS.2022.3207765

Van Belle, V., and Lisboa, P. (2014). White box radial basis function classifiers with component selection for clinical prediction models. *Artif. Intell. Med.* 60, 53–64. doi: 10.1016/j.artmed.2013.10.001

Vapnik, V. N. (1999). An overview of statistical learning theory. *IEEE Trans. Neural Netw.* 10, 988–999. doi: 10.1109/72.788640

Yazdani, A., Lu, L., Raissi, M., and Karniadakis, G. E. (2020). Systems biology informed deep learning for inferring parameters and hidden dynamics. *PLoS Comput. Biol.* 16, 1–20. doi: 10.1371/journal.pcbi.1007575

Yeh, I.-C. (1998). Modeling of strength of high-performance concrete using artificial neural networks. *Cement Concr. Res.* 28, 1797–1808. doi: 10.1016/S0008-8846(98)00165-3

Zong, T., Li, J., and Lu, G. (2021). Auxiliary model-based multi-innovation PSO identification for Wiener–Hammerstein systems with scarce measurements. *Eng. Appl. Artif. Intell.* 106:104470. doi: 10.1016/j.engappai.2021.104470