Gaussian process learning of nonlinear dynamics

Scientific Machine Learning Workshop

Dongwei Ye, Mengwu Guo
d.ye-1@utwente.nl

Mathematics of Imaging & AI, Department of Applied Mathematics, University of Twente

6th December, 2023

Mathematics of Imaging & AI

UNIVERSITY OF TWENTE.

Introduction

UNIVERSITY OF TWENTE.

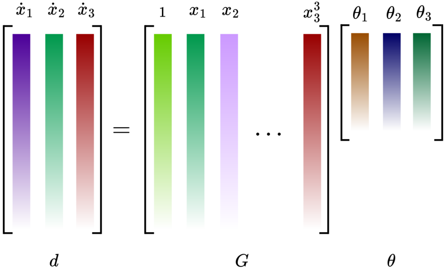
- Data-driven learning of dynamical systems from time series is an important component in scientific machine learning, as it bridges the gap between data-driven approximation and physics-based modeling

$$\begin{cases} \dot{x}_1(t) = f_1(\boldsymbol{x}(t); \boldsymbol{\theta}_1) \\ \dot{x}_2(t) = f_2(\boldsymbol{x}(t); \boldsymbol{\theta}_2) \\ \cdots \quad \cdots \\ \dot{x}_N(t) = f_N(\boldsymbol{x}(t); \boldsymbol{\theta}_N) \end{cases} \quad \text{with} \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0 \,, \quad t \geq t_0 \,.$$

Introduction

- **Identification** and **estimation**:

Sparse identification for Nonlinear Dynamics (SINDy)*



$$\boldsymbol{\theta} = \text{argmin}_{\hat{\boldsymbol{\theta}}} \frac{1}{2}||\boldsymbol{d} - \boldsymbol{G}\boldsymbol{\theta}||_2^2 + R(\boldsymbol{\theta})$$

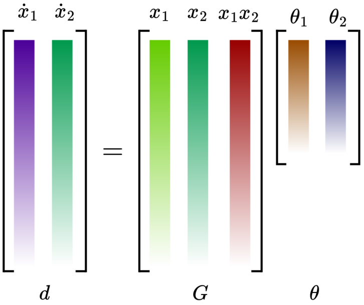
*Brunton SL, Proctor JL, Kutz JN. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. Proceedings of the national academy of sciences. 2016 Apr 12;113(15):3932-7.

Introduction

UNIVERSITY OF TWENTE.

- **Identification** and **estimation**

Operator Inference*



Lotka-Volterra

$$\begin{cases} \dot{x}_1 = \alpha x_1 - \beta x_1 x_2 \\ \dot{x}_2 = \delta x_1 x_2 - \gamma x_2 \end{cases}$$

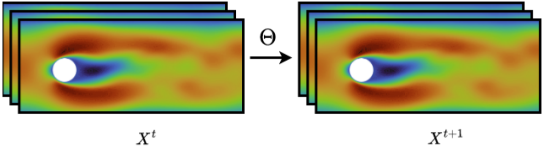
$$\boldsymbol{\theta} = \mathrm{argmin}_{\hat{\boldsymbol{\theta}}} \frac{1}{2} ||\boldsymbol{d} - \boldsymbol{G}\boldsymbol{\theta}||_2^2 + R(\boldsymbol{\theta})$$

*Peherstorfer B, Willcox K. Data-driven operator inference for nonintrusive projection-based model reduction. Computer Methods in Applied Mechanics and Engineering. 2016 Jul 1;306:196-215.

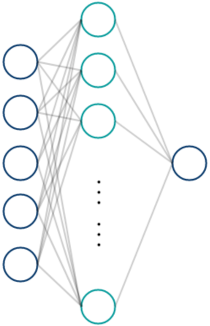
UNIVERSITY OF TWENTE.

- **Approximation**

dynamics mode decomposition*



NeuralODE**



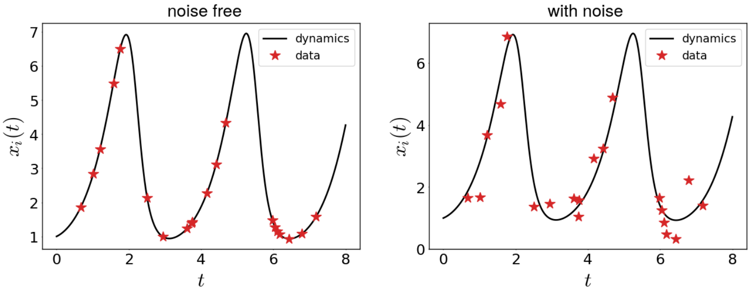
$$\frac{dx}{dt} = f(x(t), \theta)$$

*Schmid PJ. Dynamic mode decomposition of numerical and experimental data. Journal of fluid mechanics. 2010 Aug;656:5-28.

**Chen RT, Rubanova Y, Bettencourt J, Duvenaud DK. Neural ordinary differential equations. Advances in neural information processing systems. 2018;31.

Motivation

- Methods such as SINDy or Operator Inference require time derivative of the state data, which is generally not available.
- However, the predictive performance of these dynamics learning techniques may be compromised when data are scarce and/or corrupted by noise.



- Could we also enable uncertainty quantification?

Gaussian process

UNIVERSITY OF TWENTE.

- GP: a stochastic process (a collection of random variables), any finite number of which have a joint Gaussian distribution, e.g.:

$$x_i(t) \sim \mathcal{GP}(0, \kappa_i(t, t')) \,.$$

- Observations $\{t_j, x_i(t_j)\}_{j=1}^{T} = \{\mathcal{T}, X_i(\mathcal{T})\}$
- Interpolation/regression

$$\begin{bmatrix} x_i(\mathcal{T}) \\ x_i(t^*) \end{bmatrix} \sim \mathcal{GP}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \kappa_i(\mathcal{T},\mathcal{T}) & \kappa_i(\mathcal{T},t^*) \\ \kappa_i(t^*,\mathcal{T}) & \kappa_i(t^*,t^*) \end{bmatrix} \right)$$

$$\Rightarrow x_i(t^*)|t^*, \mathcal{T}, X_i(\mathcal{T}) \sim \mathcal{N}(\kappa_i(t^*,\mathcal{T})\kappa_i(\mathcal{T},\mathcal{T})^{-1}X_i(\mathcal{T}),\\ \kappa_i(t^*,t^*) - \kappa_i(t^*,\mathcal{T})\kappa_i(\mathcal{T},\mathcal{T})^{-1}\kappa_i(\mathcal{T},t^*))$$

Problem and notations

$$\begin{cases} \dot{x}_1(t) = f_1(\boldsymbol{x}(t); \boldsymbol{\theta}_1) \\ \dot{x}_2(t) = f_2(\boldsymbol{x}(t); \boldsymbol{\theta}_2) \\ \cdots \quad \cdots \\ \dot{x}_N(t) = f_N(\boldsymbol{x}(t); \boldsymbol{\theta}_N) \end{cases} \quad \text{with} \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0\,, \quad t \geq t_0\,,$$

- Collect the observed data of $\{x_i(t_k)\}_{k=0}^{K-1}$ at the time-instances $\mathcal{T}$. Let $\boldsymbol{U} = [\boldsymbol{u}_1 \;\cdots\; \boldsymbol{u}_N]^\mathsf{T} = [\boldsymbol{x}(t_0) \;\cdots\; \boldsymbol{x}(t_{K-1})] \in \mathbb{R}^{N\times K}$ be the full-state data.
- $\boldsymbol{d}_i \in \mathbb{R}^K$ denote the time derivatives of x_i over $\mathcal{T}$.

Likelihood definition with a GP

$\boxed{\text{Any linear transformation of a GP such as differentiation and integration is still a GP.}}$

A vector-valued GP for likelihood:

$$\begin{bmatrix} x_i(t) \\ \dot{x}_i(t) \end{bmatrix} \sim \mathcal{GP}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \kappa_i(t,t') & \partial_{t'}\kappa_i(t,t') \\ \partial_t \kappa_i(t,t') & \partial_t \partial_{t'} \kappa_i(t,t') \end{bmatrix} \right) .$$

$$\Rightarrow p\left(\boldsymbol{d}_i = f_i(\boldsymbol{U};\boldsymbol{\theta}_i), \boldsymbol{u}_i \middle| \, \boldsymbol{\theta}_i\right) \propto \exp\left(-\frac{1}{2} \begin{bmatrix} f_i(\boldsymbol{U};\boldsymbol{\theta}_i) \\ \boldsymbol{u}_i \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \boldsymbol{K}_i^{dd} & \boldsymbol{K}_i^{du} \\ \boldsymbol{K}_i^{ud} & \boldsymbol{K}_i^{uu} \end{bmatrix}^{-1} \begin{bmatrix} f_i(\boldsymbol{U};\boldsymbol{\theta}_i) \\ \boldsymbol{u}_i \end{bmatrix} \right),$$

$$\boldsymbol{K}_i^{dd} = \partial_t \partial_{t'} \kappa_i(\mathcal{T},\mathcal{T}) + \chi_i^d \boldsymbol{I}_K \in \mathbb{R}^{K\times K},$$

$$\boldsymbol{K}_i^{uu} = \kappa_i(\mathcal{T},\mathcal{T}) + \chi_i^u \boldsymbol{I}_K \in \mathbb{R}^{K\times K},$$

$$\boldsymbol{K}_i^{du} = \partial_t \kappa_i(\mathcal{T},\mathcal{T}) = (\boldsymbol{K}_i^{ud}) \in \mathbb{R}^{K\times K}.$$

Prior and Posterior

- Prior of parameters $\boldsymbol{\theta}_i$

$$p(\boldsymbol{\theta}_i) \propto \exp\left(-\frac{\lambda_i}{\nu}\|\boldsymbol{\theta}_i\|_\nu^\nu\right) ,$$

- Posterior via Bayesian inference

$$\begin{aligned} &p\left(\boldsymbol{\theta}_i \middle| \, \boldsymbol{d}_i = f_i(\boldsymbol{U};\boldsymbol{\theta}_i), \boldsymbol{u}_i\right) \\ &\qquad \propto p\left(\boldsymbol{d}_i = f_i(\boldsymbol{U};\boldsymbol{\theta}_i), \boldsymbol{u}_i \middle| \, \boldsymbol{\theta}_i\right) p(\boldsymbol{\theta}_i) \\ &\qquad \propto \exp\left(-\frac{1}{2}\left(f_i(\boldsymbol{U};\boldsymbol{\theta}_i)^\mathsf{T}\boldsymbol{R}_i^{dd} f_i(\boldsymbol{U};\boldsymbol{\theta}_i) + 2 f_i(\boldsymbol{U};\boldsymbol{\theta}_i)^\mathsf{T}\boldsymbol{R}_i^{du}\boldsymbol{u}_i + \frac{2\lambda_i}{\nu}\|\boldsymbol{\theta}_i\|_\nu^\nu\right)\right) . \end{aligned}$$

where

$$\begin{bmatrix} \boldsymbol{K}_i^{dd} & \boldsymbol{K}_i^{du} \\ \boldsymbol{K}_i^{ud} & \boldsymbol{K}_i^{uu} \end{bmatrix}^{-1} = \begin{bmatrix} \boldsymbol{R}_i^{dd} & \boldsymbol{R}_i^{du} \\ \boldsymbol{R}_i^{ud} & \boldsymbol{R}_i^{uu} \end{bmatrix}$$

Prior and Posterior

UNIVERSITY OF TWENTE.

- Furthermore, we define an estimate of the time derivatives over $\mathcal{T}$ (i.e., $\dot{x}_i(\mathcal{T})$) using a Gaussian process regression with the state observations $(\mathcal{T}, \boldsymbol{u}_i)$ at the same instances, given as

$$\hat{\boldsymbol{d}}_i := \boldsymbol{K}_i^{du}(\boldsymbol{K}_i^{uu})^{-1}\boldsymbol{u}_i \,.$$

- Considering the fact that $\boldsymbol{R}_i^{dd}\hat{\boldsymbol{d}}_i = -\boldsymbol{R}_i^{du}\boldsymbol{u}_i$

$$\begin{aligned}
&\propto \;\exp\left(-\frac{1}{2}\left(f_i(\boldsymbol{U};\boldsymbol{\theta}_i)^\mathsf{T}\boldsymbol{R}_i^{dd} f_i(\boldsymbol{U};\boldsymbol{\theta}_i) + 2 f_i(\boldsymbol{U};\boldsymbol{\theta}_i)^\mathsf{T}\boldsymbol{R}_i^{du}\boldsymbol{u}_i + \frac{2\lambda_i}{\nu}\|\boldsymbol{\theta}_i\|_\nu^\nu\right)\right) \,. \\
&\propto \;\exp\left(-\frac{1}{2}\left(\left\| f_i(\boldsymbol{U};\boldsymbol{\theta}_i) - \hat{\boldsymbol{d}}_i\right\|^2_{\boldsymbol{R}_i^{dd}} + \frac{2\lambda_i}{\nu}\|\boldsymbol{\theta}_i\|_\nu^\nu\right)\right) \,,
\end{aligned}$$

where $\|\boldsymbol{z}\|_{\boldsymbol{R}_i^{dd}} := \sqrt{\boldsymbol{z}^\mathsf{T}\boldsymbol{R}_i^{dd}\boldsymbol{z}}$, $\boldsymbol{z} \in \mathbb{R}^K$.

Prior and Posterior

- A physically meaningful parametrization for the dynamical system is independent of initial conditions, so should the inference of parameters $\boldsymbol{\theta}_i$ be:

$$p\left(\boldsymbol{\theta}_i \middle| \, \boldsymbol{d}_i = f_i(\boldsymbol{U};\boldsymbol{\theta}_i), \boldsymbol{u}_i\right) \propto \exp\left(-\frac{1}{2}\left(\sum_{\text{ICs}} \left\| f_i(\boldsymbol{U};\boldsymbol{\theta}_i) - \hat{\boldsymbol{d}}_i \right\|^2_{\boldsymbol{R}_i^{dd}} + \frac{2\lambda_i}{\nu}\|\boldsymbol{\theta}_i\|_\nu^\nu\right)\right) .$$

- With shared parameters:

$$\begin{aligned} & p\left(\boldsymbol{\theta}_i, \boldsymbol{\theta}_j \middle| \, \boldsymbol{d}_i = f_i(\boldsymbol{U};\boldsymbol{\theta}_i), \boldsymbol{d}_j = f_j(\boldsymbol{U};\boldsymbol{\theta}_j), \boldsymbol{u}_i, \boldsymbol{u}_j\right) \\ \propto \; & \exp\left(-\frac{1}{2}\left(\left\| f_i(\boldsymbol{U};\boldsymbol{\theta}_i) - \hat{\boldsymbol{d}}_i \right\|^2_{\boldsymbol{R}_i^{dd}} + \left\| f_j(\boldsymbol{U};\boldsymbol{\theta}_j) - \hat{\boldsymbol{d}}_j \right\|^2_{\boldsymbol{R}_j^{dd}} + \frac{2\lambda_i}{\nu}\|\boldsymbol{\theta}_i\|_\nu^\nu + \frac{2\lambda_j}{\nu}\|\boldsymbol{\theta}_j\|_\nu^\nu\right)\right) . \end{aligned}$$

- Bayesian prediction by marginalizing over the posterior of parameters

$$p(\boldsymbol{x}(t)|\boldsymbol{d}_1, \boldsymbol{u}_1, \cdots, \boldsymbol{d}_N, \boldsymbol{u}_N) = \int p(\boldsymbol{x}(t)|\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_N) \prod_{i=1}^{N} p\left(\boldsymbol{\theta}_i \middle| \, \boldsymbol{d}_i = f_i(\boldsymbol{U};\boldsymbol{\theta}_i), \boldsymbol{u}_i\right) d\boldsymbol{\theta}_i \, .$$

Case I: Estimation with linear parametrization

UNIVERSITY OF TWENTE.

- Consider an affine parametrization of $f_i(\cdot;\boldsymbol{\theta}_i)$ that is linear with respect to $\boldsymbol{\theta}_i$, i.e.,

$$f_i(\boldsymbol{x};\boldsymbol{\theta}_i) = \boldsymbol{g}_i(\boldsymbol{x})^\mathsf{T}\boldsymbol{\theta}_i\,.$$

  Hence, $f_i(\boldsymbol{U};\boldsymbol{\theta}_i)$ is rewritten as $\boldsymbol{G}_i\boldsymbol{\theta}_i$ with $\boldsymbol{G}_i := [\boldsymbol{g}_i(\boldsymbol{x}(t_0)) \;\; \cdots \;\; \boldsymbol{g}_i(\boldsymbol{x}(t_{K-1}))]^\mathsf{T} \in \mathbb{R}^{K\times p_i}$
- The proposed inference method can be represented as:

$$\begin{aligned} p(\boldsymbol{\theta}_i|\ \boldsymbol{d}_i = \boldsymbol{G}_i\boldsymbol{\theta}_i, \boldsymbol{u}_i) &\propto \exp\left(-\frac{1}{2}\left(\left\|\boldsymbol{G}_i\boldsymbol{\theta}_i - \hat{\boldsymbol{d}}_i\right\|^2_{\boldsymbol{R}_i^{dd}} + \lambda_i\|\boldsymbol{\theta}_i\|_2^2\right)\right) \\ &\propto \exp\left(-\frac{1}{2}(\boldsymbol{\theta}_i - \boldsymbol{\mu}_i)^\mathsf{T}\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{\theta}_i - \boldsymbol{\mu}_i)\right)\,, \end{aligned}$$

  with mean vector:

$$\boldsymbol{\mu}_i = (\boldsymbol{\theta}_i)_{\mathsf{MAP}} = (\boldsymbol{G}_i^\mathsf{T}\boldsymbol{R}_i^{dd}\boldsymbol{G}_i + \lambda_i\boldsymbol{I}_{p_i})^{-1}\boldsymbol{G}_i^\mathsf{T}\boldsymbol{R}_i^{dd}\hat{\boldsymbol{d}}_i\,,$$

  and the posterior covariance:

$$\boldsymbol{\Sigma}_i = (\boldsymbol{G}_i^\mathsf{T}\boldsymbol{R}_i^{dd}\boldsymbol{G}_i + \lambda_i\boldsymbol{I}_{p_i})^{-1}\,.$$

Case I: Estimation with linear parametrization

UNIVERSITY OF TWENTE.

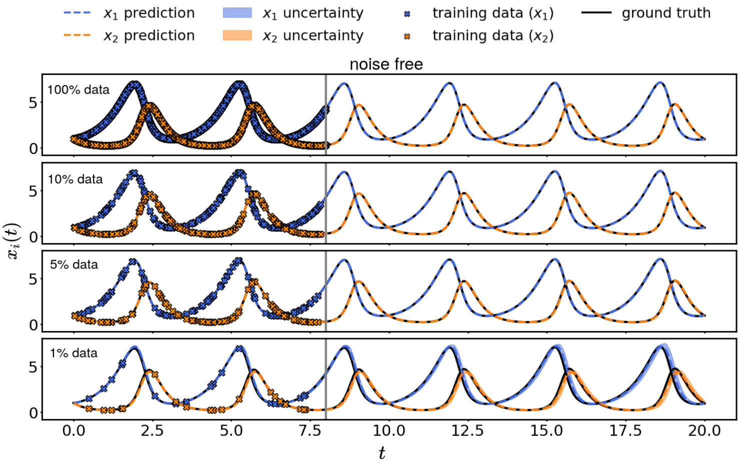
- Example: Lotka-Volterra model

$$\begin{cases} \dot{x}_1 = \alpha x_1 - \beta x_1 x_2 \,, \\ \dot{x}_2 = \delta x_1 x_2 - \gamma x_2 \,, \end{cases}$$

$$\Rightarrow \quad \begin{aligned} \boldsymbol{g}_1(\boldsymbol{x}) &= [\, x_1 \quad x_1 x_2 \,]^\mathsf{T}, \quad \boldsymbol{\theta}_1 = [\, \alpha \quad -\beta \,]^\mathsf{T}; \\ \boldsymbol{g}_2(\boldsymbol{x}) &= [\, x_1 x_2 \quad x_2 \,]^\mathsf{T}, \quad \boldsymbol{\theta}_2 = [\, \delta \quad -\gamma \,]^\mathsf{T}. \end{aligned}$$

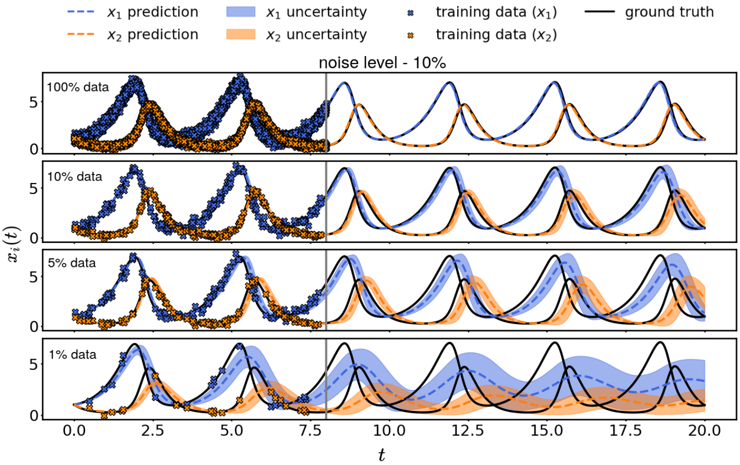
Case I: Lotka-Volterra model

UNIVERSITY OF TWENTE.



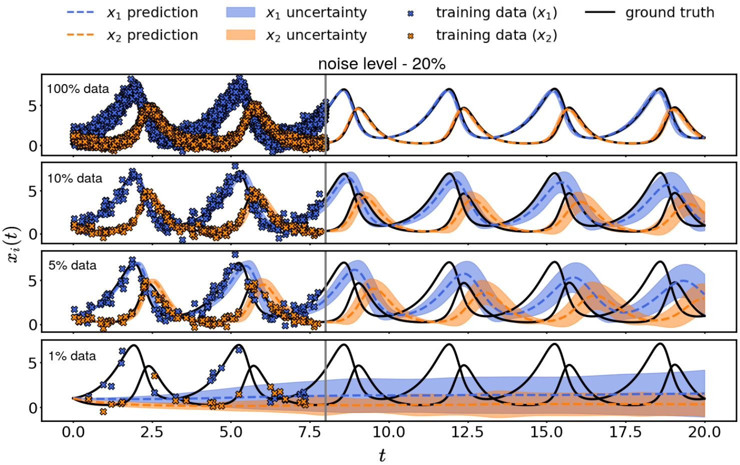
Case I: Lotka-Volterra model

UNIVERSITY OF TWENTE.



Case I: Lotka-Volterra model

UNIVERSITY OF TWENTE.



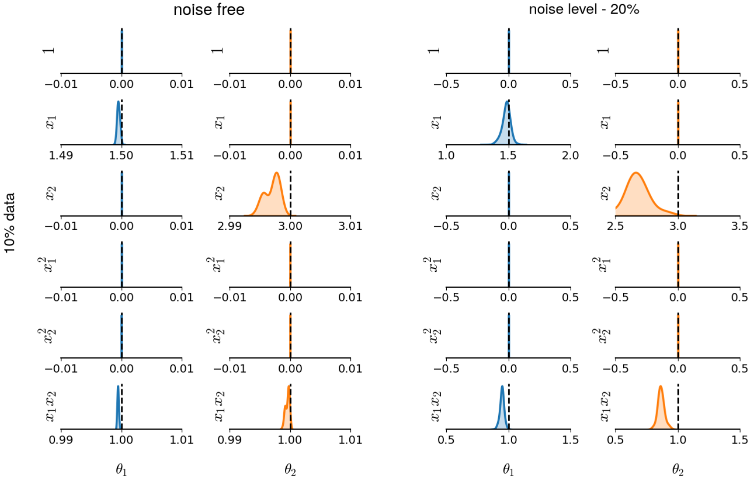
Case II: Sparse identification

- The proposed inference method can be represented as:

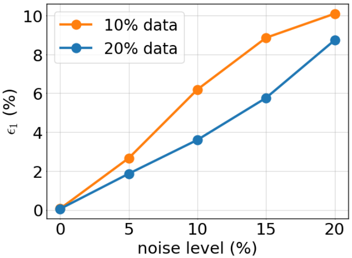
$$p(\boldsymbol{\theta}_i|\ \boldsymbol{d}_i = \boldsymbol{G}_i\boldsymbol{\theta}_i, \boldsymbol{u}_i) \propto \exp\left(-\frac{1}{2}\left(\left\|\boldsymbol{G}_i\boldsymbol{\theta}_i - \hat{\boldsymbol{d}}_i\right\|^2_{\boldsymbol{R}_i^{dd}} + 2\lambda_i\left\|\boldsymbol{\theta}_i\right\|_1\right)\right).$$

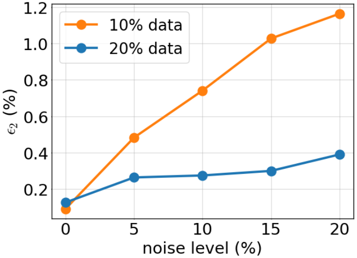
$$\boldsymbol{g}_1(\boldsymbol{x}) = [\ 1\ \ x_1\ \ x_2\ \ x_1^2\ \ x_2^2\ \ x_1x_2\ ]^\mathsf{T}, \quad \boldsymbol{\theta}_1 = [\ 0\ \ \alpha\ \ 0\ \ 0\ \ 0\ \ -\beta\ ]^\mathsf{T};$$
$$\boldsymbol{g}_2(\boldsymbol{x}) = [\ 1\ \ x_1\ \ x_2\ \ x_1^2\ \ x_2^2\ \ x_1x_2\ ]^\mathsf{T}, \quad \boldsymbol{\theta}_2 = [\ 0\ \ 0\ \ -\gamma\ \ 0\ \ 0\ \ \delta\ ]^\mathsf{T}$$

Case II: Lotka-Volterra model



Case II: Lotka-Volterra model





$$\epsilon_1 = 100\% \times \left(\frac{\sum_{i=1}^{2} \|\boldsymbol{\theta}_i - \mathbb{E}_{\text{post}}[\boldsymbol{\theta}_i]\|_2^2}{\sum_{i=1}^{2} \|\boldsymbol{\theta}_i\|_2^2} \right)^{1/2}, \epsilon_2 = 100\% \times \left(\frac{\sum_{i=1}^{2} \mathbb{V}ar_{\text{post}}[\boldsymbol{\theta}_i]}{\sum_{i=1}^{2} \|\boldsymbol{\theta}_i\|_2^2} \right)^{1/2}$$

Case III: Approximation

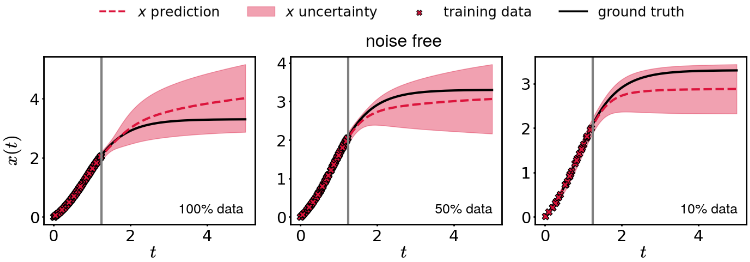
Nonlinear parametrization with a shallow neural network

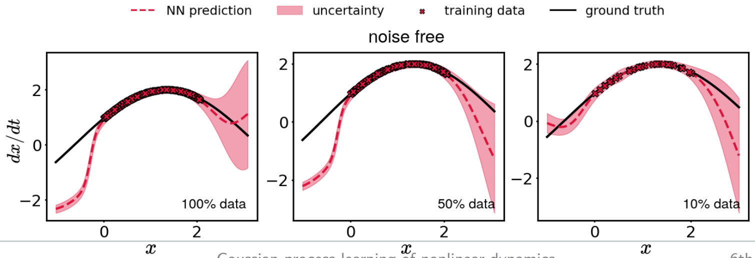
$$f_i(\boldsymbol{x};\boldsymbol{\theta}_i) = \sum_{l=1}^{L} v_{il}\ \sigma\left(\boldsymbol{w}_{il}^{\mathsf{T}}\boldsymbol{x} + b_{il}\right), \quad \text{with} \quad \boldsymbol{\theta}_i := \{v_{il}, b_{il}, \boldsymbol{w}_{il}\}_{l=1}^{L},$$

- 1D synthetic example

$$\dot{x} = \gamma \sin\left(\alpha x + \beta\right)$$

Case III


x prediction
x uncertainty
training data
ground truth
noise free
$x(t)$
t
100% data
50% data
10% data


NN prediction
uncertainty
training data
ground truth
noise free
dx/dt
x
100% data
50% data
10% data

Summary

UNIVERSITY OF TWENTE.

- Does not require a direct finite-difference estimation of time-derivatives from solution data as in OpInf or SINDy, but instead (implicitly) evaluates the derivatives Gaussian process approximation.
- Improves predictiveness and facilitates uncertainty quantification for dynamics learning with noisy and/or scarce data.
- Key features: simplicity, robustness, generality.

Thank you for your attention!

Questions?