ODIN: ODE-Informed Regression for Parameter and State Inference in Time-Continuous Dynamical Systems

Philippe Wenk,[*1,2] **Gabriele Abbati,**[*3] **Michael A Osborne,**[3]
Bernhard Schölkopf,[4] **Andreas Krause,**[1] **Stefan Bauer**[4]
[1]Learning and Adaptive Systems Group, ETH Zürich
[2]Max Planck ETH Center for Learning Systems
[3]Department of Engineering Science, University of Oxford
[4]Empirical Inference Group, Max Planck Institute for Intelligent Systems
Correspondence to: wenkph@ethz.ch, gabb@robots.ox.ac.uk

Abstract

Parameter inference in ordinary differential equations is an important problem in many applied sciences and in engineering, especially in a data-scarce setting. In this work, we introduce a novel generative modeling approach based on constrained Gaussian processes and leverage it to build a computationally and data efficient algorithm for state and parameter inference. In an extensive set of experiments, our approach outperforms the current state of the art for parameter inference both in terms of accuracy and computational cost. It also shows promising results for the much more challenging problem of model selection.

Introduction

Ordinary differential equations (ODEs) represent a ubiquitous tool for modeling problems in many quantitative sciences and engineering. While with first principles and expert knowledge it is often possible to work out a parametric form for the equations that model a system of interest, in most cases there is no closed-form solution, making parameter identification problematic. One needs to rely on numerical schemes, which can be sub-optimal given that the exact system trajectory is usually unknown: typically, we observe noisy measurements of the true trajectory only at some discrete time points. This problem has been extensively studied in the past with classical approaches based on numerical integration (e.g. Bard (1974) and Benson (1979)), also in the relevant context of model selection (Chen, Shojaie, and Witten 2017; Wu et al. 2014).

Classical approaches iteratively propose new sets of parameters and then evaluate them by numerical integration: the estimated trajectory can then be compared against the observed data. Among others, Varah (1982) argue that this procedure can be turned on its head to improve computational performance. In principle, finding good parameters is equivalent to denoising the states, since adequate ODE parameters will lead to a trajectory that is close to ground truth. In particular, Varah (1982) first fit a spline curve to the observations to approximate the true trajectory, and subsequently match state and derivative estimates of said splines to obtain the ODE parameters. This idea gave rise to a class of *gradient matching* algorithms that rely on spline regression, kernel regression and, in a Bayesian setting, Gaussian process regression (GPR).

Gaussian processes (GPs) provide a very natural and theoretically appealing way to smooth time series, especially because they are very closely related to Kalman filtering (Hartikainen and Särkkä 2010). Thus, there has been significant interest in incorporating them into the gradient matching framework, starting from the pioneering theoretical work of Calderhead, Girolami, and Lawrence (2009): they propose a GP-based probabilistic modeling scheme on which they perform inference using MCMC. Dondelinger et al. (2013) change this probabilistic setup to achieve a more efficient MCMC sampling procedure (AGM - Adaptive Gradient matching), while Gorbach, Bauer, and Buhmann (2017) introduce a computationally efficient inference scheme based on variational inference (VGM - Variational Gradient Matching). Crucially, all these methods rely on a product of experts (PoE) heuristic (an alternative approach is formulated by Barber and Wang (2014), later questioned by Macdonald, Higham, and Husmeier (2015)). However, Wenk et al. (2018) show that the PoE leads to theoretical issues: indeed, in the graphical models proposed by Calderhead, Girolami, and Lawrence (2009), Dondelinger et al. (2013) and Gorbach, Bauer, and Buhmann (2017), the ODE parameters become statistically independent of the observations. Thus, Wenk et al. (2018) propose a new graphical model that circumvents this issue and present an efficient MCMC-based inference scheme (FGPGM - Fast Gaussian Process based Gradient Matching). A further formulation that is based on variational inference and allows for additional inequality constraints on the derivatives is provided by Lorenzi and Filippone (2018).

Similarly to Gaussian process-based gradient matching, González, Vujačić, and Wit (2014) and Niu et al. (2016) use kernel ridge regression in a frequentist setting (RKG2/RKG3 - Reproducing Kernel based Gradient Matching). Aiming directly for point estimates of the parameters, their approaches are naturally faster than alternatives that build on the use of MCMC and Gaussian processes.

[*]Equal contribution

Nevertheless, they rely on several trade-off parameters to be tuned via cross-validation, which can turn out to be impractical to data-scarce environments.

In our work, we extend and blend both Bayesian and frequentist viewpoints to obtain a computationally efficient algorithm that can learn states and parameters in a low-data setting. In particular, we

- Present a novel generative model, rephrasing the parameter inference problem as constrained Gaussian process regression;
- Provide a data-efficient algorithm that concurrently estimates states and parameters;
- Show how all hyperparameters can be learned from data and how they can be used as an indicator for model mismatch;
- Provide an efficient Python implementation for public use, with a publicly available code base at https://github.com/gabb7/ODIN

Background

Problem Setting

Throughout this work, we consider K-dimensional dynamical systems whose evolution is described by a set of differential equations parameterized by a vector $\boldsymbol{\theta}$, i.e.

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

The system is observed under additive, zero-mean Gaussian noise at N discrete time points $\mathbf{t} = [t_1, \dots, t_N]$. We assume the standard deviation of the noise to be constant over time but it may differ for each state dimension. The noise is further assumed to be uncorrelated across dimensions and time points. This leads to the following observation model:

$$y_k(t_i) = x_k(t_i) + \epsilon_k(t_i), \qquad \epsilon_k(t_i) \sim \mathcal{N}(0, \sigma_k). \tag{2}$$

Previous research efforts propose to model $\mathbf{f}$ itself as a Gaussian process (e.g. Heinonen et al. (2018)) or to extend the framework to SDEs (e.g. Ryder et al. (2018), Abbati et al. (2019)). While it might be interesting to investigate how these frameworks could be combined with ODIN, this falls outside the scope of this paper. We thus restrict ourselves explicitly to the case where we have deterministic differential equations with known parametric form.

Temporal Regression

In the context of dynamical systems, Gaussian processes are employed mostly to model directly the system dynamics (i.e. the function $\mathbf{f}$). GP-based gradient matching approaches the problem differently. Here GPs model the states $\mathbf{x}$ and provide an approximation for the function $\mathbf{x}$ that maps a time point t_i to the corresponding state vector $\mathbf{x}(t_i)$. For the sake of readability, we assume $K = 1$ and denote the values of $x(t)$ stacked across time points as $\mathbf{x} = [x(t_1), \dots, x(t_N)]$. As shown in the experiment section, the extension to $K > 1$ is straightforward: K independent Gaussian processes can be stacked to model each state independently.

While our method can theoretically work with any nonlinear, differentiable regression technique, we choose to use Gaussian processes. GPs have superb analytical properties (Rasmussen and Williams 2006) and they recently showed remarkable empirical results in the context of parameter inference for ODEs (Lorenzi and Filippone 2018; Wenk et al. 2018). Moreover, thanks to the representer theorem (Schölkopf, Herbrich, and Smola 2001), Gaussian processes are closely related to kernel ridge regression: this connects GP-based gradient matching approaches to the reproducing-kernel-based ones (e.g. Niu et al. (2016)).

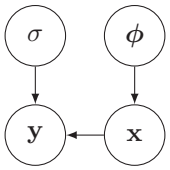


Figure 1: Generative model for standard Gaussian process regression. Given kernel hyperparameters $\boldsymbol{\phi}$ and observation noise standard deviation σ, the probability densities for the states $\mathbf{x}$ and their noisy observations $\mathbf{y}$ are fully determined.

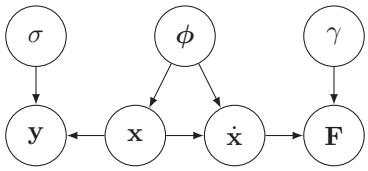


Figure 2: Generative model for GP regression with derivative observations $\mathbf{F}$, for which we use a Gaussian observation model with variance γ (as in ODIN). Due to the GP prior, $\mathbf{x}$ and $\dot{\mathbf{x}}$ are jointly Gaussian with known probability densities once the kernel hyperparameters $\boldsymbol{\phi}$ are determined.

Gaussian Process Regression

As in standard GP regression, we start by choosing a covariance function (or kernel) $k_{\boldsymbol{\phi}}$, which is parameterized by a set of hyperparameters $\boldsymbol{\phi}$. The kernel is used to compute a covariance matrix $\mathbf{C}_{\boldsymbol{\phi}}$, whose elements are given by $[\mathbf{C}_{\boldsymbol{\phi}}]_{i,j} = k_{\boldsymbol{\phi}}(t_i, t_j)$. $\mathbf{C}_{\boldsymbol{\phi}}$ can be used to define a zero-mean prior over the true states $\mathbf{x}$ at the observation times $\mathbf{t}$:

$$p(\mathbf{x} \mid \boldsymbol{\phi}) = \mathcal{N}(\mathbf{x} \mid \mathbf{0}, \mathbf{C}_{\boldsymbol{\phi}}). \tag{3}$$

The noise model from Equation (2) yields a Gaussian likelihood for the observations $\mathbf{y}$:

$$p(\mathbf{y} \mid \mathbf{x}, \sigma) = \mathcal{N}(\mathbf{y} | \mathbf{x}, \sigma^2 \mathbf{I}). \tag{4}$$

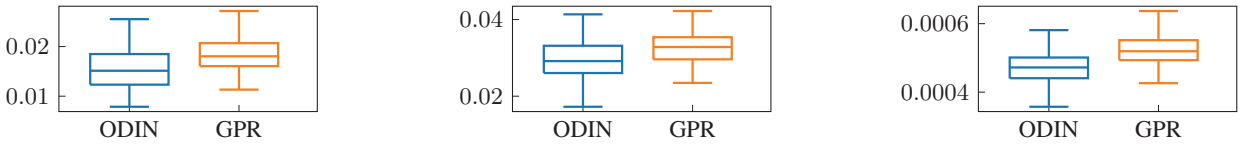
Using Bayes rule and observing the fact that a product of two Gaussians in the same variables is again a Gaussian, we obtain the classic GP posterior

$$p(\mathbf{x} \mid \mathbf{y}, \sigma, \boldsymbol{\phi}) = \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{x}}), \tag{5}$$

$$\text{where} \qquad \boldsymbol{\mu}_{\mathbf{x}} = \mathbf{C}_{\boldsymbol{\phi}}(\mathbf{C}_{\boldsymbol{\phi}} + \sigma^2 \mathbf{I})^{-1} \mathbf{y} \tag{6}$$

$$\text{and} \qquad \boldsymbol{\Sigma}_{\mathbf{x}} = \sigma^2(\mathbf{C}_{\boldsymbol{\phi}} + \sigma^2 \mathbf{I})^{-1} \mathbf{C}_{\boldsymbol{\phi}} \tag{7}$$

A graphical representation of this generative model can be found in Figure 1. Throughout this paper, we assume that $k_{\boldsymbol{\phi}}$ is *differentiable* w.r.t. both of its arguments.



(a) Lotka-Volterra (b) FitzHugh-Nagumo (c) Protein Transduction

Figure 3: RMSE of state estimates using vanilla GP regression and ODIN on the benchmark systems (low noise case). While GPR can only access the noisy observations $\mathbf{y}$, ODIN considers the parametric form of $\mathbf{f}(\mathbf{x}, \boldsymbol{\theta})$ (with no information about $\boldsymbol{\theta}$). This additional regularization contributes towards more accurate estimations.

Gaussian Process Regression with Derivatives

As previously noted e.g. by Solak et al. (2003), the estimate of the posterior distribution of the states given by Equation (5) can be further refined if we consider access to noisy observations of the derivatives. Let us then assume we have additional observations $\mathbf{F}$ that are generated by $F(t_i) = \dot{x}(t_i) + \delta(t_i)$, where $\delta(t_i) \sim \mathcal{N}(0, \gamma)$. Incorporating such derivatives is straightforward: since Gaussian processes are closed under linear operations, the distribution over the derivatives is again a Gaussian process. Following the notation of Wenk et al. (2018) and the argumentation in their appendix, we obtain

$$p(\dot{\mathbf{x}} \mid \mathbf{x}, \boldsymbol{\phi}) = \mathcal{N}(\dot{\mathbf{x}} \mid \mathbf{D}\mathbf{x}, \mathbf{A}), \tag{8}$$

where the exact form of the matrices $\mathbf{D}$ and $\mathbf{A}$ is omitted for simplicity, but can be found in the supplementary material. Equation (8) can now be combined with the likelihood for the derivative observations

$$p(\mathbf{F} \mid \dot{\mathbf{x}}, \gamma) = \mathcal{N}(\mathbf{F} \mid \dot{\mathbf{x}}, \gamma\mathbf{I}). \tag{9}$$

This leads to the generative model shown in Figure 2. Just like in standard Gaussian process regression, all posteriors of interest can be calculated analytically, since all probability densities are Gaussian distributions in $\mathbf{x}$, $\dot{\mathbf{x}}$ or linear transformations thereof.

ODE-Informed Regression

Gaussian Process-based Gradient Matching

Given the model in Figure 2, the main challenge consists in including the mathematical expressions of the ODEs in a meaningful way. In traditional GP-based gradient matching, the ODEs are introduced as a second generative model for $\mathbf{F}$ or $\dot{\mathbf{x}}$. The latter is then combined with the Gaussian process model of Figure 2 to establish a probabilistic link between the observations $\mathbf{y}$ and the parameters $\boldsymbol{\theta}$. However, the GP model fully determines the probability densities of $\mathbf{F}$ and $\dot{\mathbf{x}}$. Thus, the two generative models have to be combined using some heuristic, like the product of experts (Calderhead, Girolami, and Lawrence 2009; Dondelinger et al. 2013; Gorbach, Bauer, and Buhmann 2017) or an additional Dirac delta function forcing equality (Wenk et al. 2018).

The resulting, unified generative model is then used to approximate the posterior of $\mathbf{x}$ and $\boldsymbol{\theta}$ through Bayesian inference techniques, e.g. MCMC (Calderhead, Girolami, and Lawrence 2009; Dondelinger et al. 2013; Wenk et al. 2018) or variational mean field (Gorbach, Bauer, and Buhmann 2017). Inference for these algorithms consists in computing mean and standard deviation of an approximate posterior to get estimates that include uncertainty. As we shall see in the experiment section, this works well for sufficiently tame dynamics and identifiable systems, but struggles to produce meaningful results for multi-modal posteriors. Crucially, practical systems often produce multi-modal posteriors and suffer from unidentifiability without strong priors (e.g. Stephan et al. (2008), Hass et al. (2017)). As we shall see, ODE-informed regression can overcome this issue.

ODIN: ODE-Informed Regression

To avoid the problems associated with the two probabilistic models of traditional GP-based gradient matching, ODIN does not include the ODEs via a separate generative model. Instead, they are introduced at inference time in the form of *constraints*, essentially solving a constrained MAP problem.

We start with the joint density of the Gaussian process described in Figure 2, denoted by $p(\mathbf{y}, \mathbf{x}, \dot{\mathbf{x}}, \mathbf{F} \mid \sigma, \gamma, \boldsymbol{\phi})$. As a result of the Gaussian observation model for $\mathbf{F}$, $\dot{\mathbf{x}}$ can be marginalized out analytically, leading to

$$p(\mathbf{y}, \mathbf{x}, \mathbf{F} \mid \sigma, \gamma, \boldsymbol{\phi}) = \tag{10}$$
$$\mathcal{N}(\mathbf{y} \mid \mathbf{x}, \sigma^2\mathbf{I})\, \mathcal{N}(\mathbf{x} \mid \mathbf{0}, \mathbf{C}_{\boldsymbol{\phi}})\, \mathcal{N}(\mathbf{F} \mid \mathbf{D}\mathbf{x}, \mathbf{A} + \gamma\mathbf{I}).$$

Assuming fixed values for $\sigma, \boldsymbol{\phi}$ and γ, this equation can be simplified by taking the logarithm, discarding all terms that do not explicitly depend on the states $\mathbf{x}$ and the derivative observations $\mathbf{F}$ and ignoring multiplicative factors to obtain

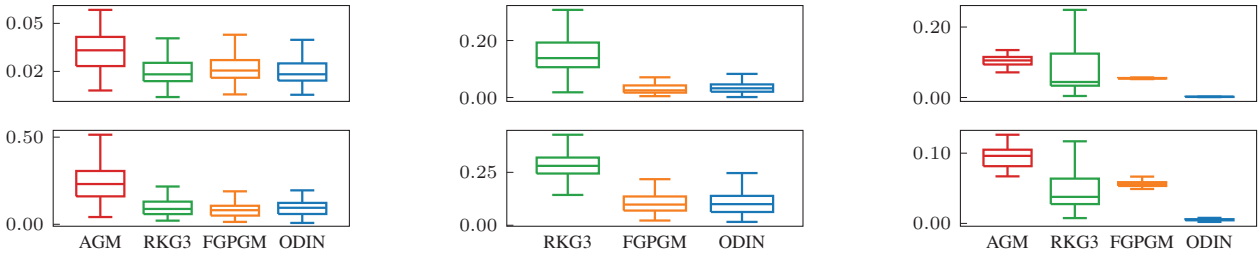
$$\tilde{\mathcal{R}}(\mathbf{x}, \mathbf{F}, \mathbf{y}) = \tag{11}$$
$$||\mathbf{x}||^2_{\mathbf{C}_{\boldsymbol{\phi}}^{-1}} + ||\mathbf{x} - \mathbf{y}||^2_{\sigma^{-2}\mathbf{I}} + ||\mathbf{F} - \mathbf{D}\mathbf{x}||^2_{(\mathbf{A}+\gamma\mathbf{I})^{-1}},$$

where $||\mathbf{u}||^2_{\mathbf{M}} := \mathbf{u}^T\mathbf{M}\mathbf{u}$ is the norm of the vector $\mathbf{u}$ weighted by a positive-definite matrix $\mathbf{M}$.

The key mechanism behind ODIN lies in how we obtain values for $\mathbf{F}$. In principle, $\mathbf{F}$ could be marginalized out to recover standard GP regression. However, this is not desirable, as we would ignore the ODE information. Instead, ODIN includes the ODEs as additional constraints in the optimization problem: rather than keeping $\mathbf{F}$ completely flexible, we assume the existence of a parameter vector $\boldsymbol{\theta}$ that links the derivative observations to the ODEs. More formally,

$$\mathbf{x}, \mathbf{F} = \arg\min_{\mathbf{x}, \boldsymbol{\theta}} \tilde{\mathcal{R}}(\mathbf{x}, \mathbf{F}, \mathbf{y}) \tag{12}$$

$$\text{s. t.} \quad \exists\boldsymbol{\theta} \quad \text{with} \quad \mathbf{f}(\mathbf{x}(t_i), \boldsymbol{\theta}) = \mathbf{F}_i \quad \text{for all } i. \tag{13}$$



(a) Lotka-Volterra (b) FitzHugh-Nagumo (c) Protein Transduction

Figure 4: Trajectory RMSE for parameter inference on the benchmark systems. The top row shows the low noise case with $\sigma = 0.1$ for LV, $SNR = 100$ for FHN and $\sigma = 0.001$ for PT. The bottom row shows the high noise case with $\sigma = 0.5$ for LV, $SNR = 10$ for FHN and $\sigma = 0.01$ for PT.

As it turns out, these constraints can be incorporated in the optimization problem by directly substituting $\mathbf{F}$ with the corresponding contribution from the ODEs, leading to

$$\mathbf{x}, \boldsymbol{\theta} = \arg\min_{\mathbf{x},\boldsymbol{\theta}} \mathcal{R}(\mathbf{x}, \boldsymbol{\theta}, \mathbf{y}), \tag{14}$$

where $\mathcal{R}(\mathbf{x}, \boldsymbol{\theta}, \mathbf{y}) := \tilde{\mathcal{R}}(\mathbf{x}, \boldsymbol{f}(\mathbf{x}, \boldsymbol{\theta}), \mathbf{y})$.

This constitutes the key idea behind ODIN. Instead of providing direct observations of the derivatives, we generate them via the ODEs after fixing the ODE parameters $\boldsymbol{\theta}$.

Similarly to classical frequentist methods (Varah 1982; Niu et al. 2016), $\mathcal{R}(\mathbf{x}, \boldsymbol{\theta}, \mathbf{y})$ punishes divergence between the states $\mathbf{x}$ and the observations $\mathbf{y}$, as well as between the output of the ODEs, $\mathbf{f}(\mathbf{x}, \boldsymbol{\theta})$, and the derivatives estimated by the regressing GP; the regularization term avoids overfitting. However, in sharp contrast to frequentist approaches, all trade-off parameters are naturally provided by the GP framework, once the hyperparameters $\boldsymbol{\phi}$ and noise levels σ and γ are fixed. As we will see later, the absence of cross-validation for hyperparameter learning crucially improves accuracy in sparsely observed systems.

Algorithm 1 ODIN

Input: $\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(K)}, \mathbf{f}(\mathbf{x}, \boldsymbol{\theta})$
Step 1: State-independent GP regression
for all $k \in K$ **do**
  Standardize time $\mathbf{t}$ and observations $\mathbf{y}_k$.
  Fit $\boldsymbol{\phi_k}$ and σ_k using empirical Bayes, i.e. maximize $p(\mathbf{y}^{(k)}|\mathbf{t}, \boldsymbol{\phi}_k, \sigma_k)$.
  Initialize $\mathbf{x}_k$ using the mean $\boldsymbol{\mu}_k$ of the trained GP.
end for
Step 2: ODE Information Incorporation
Initialize $\boldsymbol{\theta}$ randomly.
Initialize $\gamma_1, \ldots, \gamma_K = 1.0$
Apply L-BFGS-B to solve the optimization problem (15) and obtain $\hat{\mathbf{x}}, \hat{\boldsymbol{\theta}}, \hat{\gamma}_1, \ldots, \hat{\gamma}_K$.
Return: $\hat{\mathbf{x}}, \hat{\boldsymbol{\theta}}, \hat{\gamma}_1, \ldots, \hat{\gamma}_K$

Derivative Observation Model

Let us recall that we conceptually substitute the ODE outputs with derivative observations that are subjected to Gaussian noise, with variance γ. In this process, we can interpret in an intuitive but meaningful way: during and after training, the ODE outputs and the GP derivative estimates can deviate from the ground truth and thus differ from each other. This divergence is accounted for by γ. In classical GP-based approaches (Dondelinger et al. 2013; Gorbach, Bauer, and Buhmann 2017; Wenk et al. 2018), γ is treated as a random variable whose values are independent of the inference procedure; sometimes it is fixed a priori (Gorbach, Bauer, and Buhmann 2017; Wenk et al. 2018). However, we can expect that the divergence between ODEs and GP derivatives would be larger in the early steps of training, while it should decrease when the ODEs describe well to the ground truth. Thus, it is sensible to automatically adapt γ to reflect the current quality of the estimates.

The ODIN framework can be adjusted to reflect this reasoning. To obtain Equation (11), we implicitly assumed γ to be constant. If we rather treat it as an optimization parameter, the objective of Equation (14) changes to

$$\mathbf{x}, \boldsymbol{\theta}, \gamma = \arg\min_{\mathbf{x},\boldsymbol{\theta},\gamma} \mathcal{R}(\mathbf{x}, \boldsymbol{\theta}, \mathbf{y}, \gamma) \tag{15}$$

where

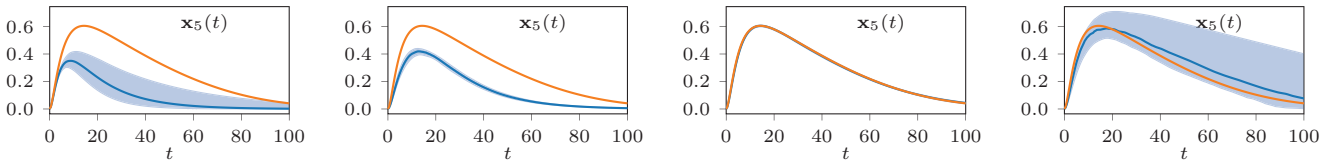
$$\mathcal{R}(\mathbf{x}, \boldsymbol{\theta}, \mathbf{y}, \gamma) = \mathcal{R}(\mathbf{x}, \boldsymbol{\theta}, \mathbf{y}) + \log(\det(\mathbf{A} + \gamma\mathbf{I})). \tag{16}$$

If γ is part of the optimization procedure, the contribution of the normalization constant in Equation (10) can not be ignored when deriving the risk appearing in Equation (16). In practice, similarly to the log-determinant in standard GP regression, this term acts as an Occam's razor by preventing an excessive growth of γ if the GP derivatives and the ODE outputs differ significantly.

The final ODIN routine is summarized as Algorithm 1.

Remarks

Throughout this work, we assume to have access to observations $\mathbf{y}$ that are subjected to the noise model described in Equation (2). However, the Gaussian noise assumption is only needed when deriving the term $||\mathbf{x} - \mathbf{y}||^2_{\sigma^{-2}\mathbf{I}}$ in $\mathcal{R}$. Thus, it could be straightforward to accommodate for alternative noise models by adjusting the corresponding term in the risk formula. On the other side, a Gaussian noise model (with variance γ) is a strict requirement for the derivative observations, as it is necessary to marginalize $\dot{\mathbf{x}}$ analytically.



(a) AGM (b) FGPGM (c) ODIN (d) RKG3

Figure 5: Comparison of the trajectories obtained by numerically integrating the inferred parameters of the Protein Transduction system for $\sigma = 0.01$. The solid blue line is the median trajectory, while for clarity we shaded the area between the 25% and 75% quantiles. The orange trajectory represents the ground truth.

As demonstrates in the experiments section, in the case of perfect ODEs (i.e. when we know the true parametric form a priori), γ can in principle be set to zero; nevertheless, when that is not the case it provides an effective mechanism for detecting model mismatch and helps with the challenging problem of model selection.

Experiments

In this section, we demonstrate the versatility of ODIN and compare its performance to various state-of-the-art methods. We start by comparing its parameter inference capabilities to various state-of-the-art inference schemes on three commonly used benchmark systems: the Lotka-Volterra (LV) predator-prey model (Lotka 1932); the FitzHugh-Nagumo (FHN) neuronal model (FitzHugh 1961; Nagumo, Arimoto, and Yoshizawa 1962); the chemical protein transduction (PT) system as seen in Vyshemirsky and Girolami (2007). All three systems have already been studied extensively in the context of gradient matching (Calderhead, Girolami, and Lawrence 2009; Dondelinger et al. 2013; Gorbach, Bauer, and Buhmann 2017; Wenk et al. 2018) and thus represent a clear benchmark. For completeness, we restate the concrete parametric form in the supplementary material, together with the ground truth for all parameters. In addition to state and parameter inference, we show how ODIN can be used for model selection, a missing feature for every comparison method here considered. Finally, we prove linear scaling behavior of ODIN in the state dimension K by investigating its performance on a high-dimensional, fourth benchmark system with up to 1000 states.

Evaluation Details and Data Creation

All experimental datasets are generated using numerical simulations. Thus, the ground truth for both the states $\mathbf{x}^*$ and parameters $\boldsymbol{\theta}^*$ is always available. Following Wenk et al. (2018), we employ the trajectory RMSE as a metric to compare the quality of parameter estimates. For ease of reference, we restate the definition in Definition 1.

Definition 1 (Trajectory RMSE) *Let $\hat{\boldsymbol{\theta}}$ be the parameters estimated by an inference algorithm. Let $\mathbf{t}$ be the vector collecting the observation times. Define $\tilde{x}(t)$ as the trajectory one obtains by integrating the ODEs using the estimated parameters, but the true initial value, i.e.*

$$\tilde{x}(0) = \mathbf{x}^*(0) \tag{17}$$

$$\tilde{x}(t) = \int_0^t f(\tilde{x}(s), \hat{\boldsymbol{\theta}}) ds \tag{18}$$

and define $\tilde{\mathbf{x}}$ element-wise as its evaluation at observation times $\mathbf{t}$, i.e. $\tilde{x}_i = \tilde{x}(t_i)$. The trajectory RMSE is then defined as

$$\textit{tRMSE} := \frac{1}{N} ||\tilde{\mathbf{x}} - \mathbf{x}||_2, \tag{19}$$

where $||.||_2$ denotes the standard Euclidean 2-norm.

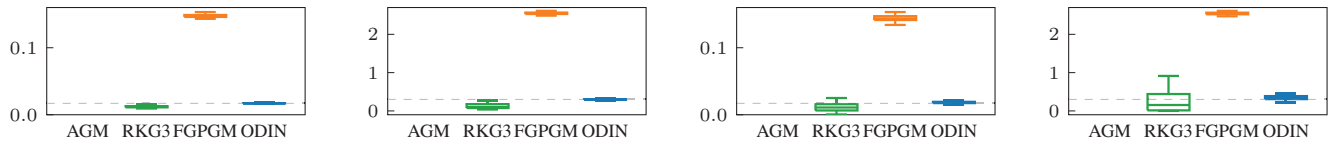
To evaluate the robustness of each algorithm w.r.t. different observation noise realizations, we always run 100 repetitions for every experimental setting. In each repetition, we keep $\mathbf{x}^*$ and $\boldsymbol{\theta}^*$ fixed and only sample the noise on $\mathbf{y}$. Results are then reported as quantiles over these 100 runs.

State and Parameter Inference

In the parameter inference setting, the true parametric form of the dynamical system is assumed to be provided by a practitioner, derived through first principles or expert knowledge. Thus, together with the noisy observations $\mathbf{y}$, we have access to the true parametric form $\dot{\mathbf{x}} = f(\mathbf{x}, \boldsymbol{\theta})$. The goal is to recover the true states $\mathbf{x}$ and parameters $\boldsymbol{\theta}^*$ at observation time. While smoothing is important, estimating $\boldsymbol{\theta}^*$ is of greater practical importance.

Out of the three comparison algorithms we chose, AGM (Dondelinger et al. 2013) and FGPGM (Wenk et al. 2018) rely on Gaussian processes and MCMC inference, while RKG3 (Niu et al. 2016) chooses a frequentist, kernel-regression-based approach. For all comparisons, implementations provided by the respective authors are used. Once more in accordance to the gradient matching literature, evaluations include both a low and a high noise setting for every system.

As shown by Solak et al. (2003), including direct observations of $\mathbf{F}$ can improve the accuracy of GP regression. ODIN does not have access to such observations, but it leverages the parametric form of the ODEs as a regularizer when performing state inference. As can be seen in Figure 3, this regularization actually improves the estimates of the states. This fact motivates a key difference to (Calderhead, Girolami, and Lawrence 2009), who propose to first fit the states using GPR and then perform gradient matching while keeping the states fixed. In the following, we show how



(a) $\theta_5, \sigma = 0.001$ (b) $\theta_6, \sigma = 0.001$ (c) $\theta_5, \sigma = 0.01$ (d) $\theta_6, \sigma = 0.01$

Figure 6: Parameter estimates for Protein Transduction for $\sigma = 0.001$ (a-b) and $\sigma = 0.01$ (c-d). Showing median, 50% and 75% quantiles over 100 independent noise realizations. The dashed line indicates the ground truth.

Table 1: Median and standard deviation of computation time (in seconds) for parameter inference over 100 independent noise realizations.

	AGM$[s]$	RKG3$[s]$	FGPGM$[s]$	ODIN$[s]$
LV, $\sigma = 0.1$	4548.0 ± 453.8	79.0 ± 19.0	3169.5 ± 90.1	$\mathbf{13.4 \pm 5.1}$
LV, $\sigma = 0.5$	4545.0 ± 558.5	76.5 ± 15.8	3187.5 ± 340.9	$\mathbf{11.4 \pm 5.1}$
FHN, $SNR = 100$	/	74.5 ± 14.3	8678.0 ± 482.7	$\mathbf{5.8 \pm 3.5}$
FHN, $SNR = 10$	/	77.5 ± 12.3	8677.0 ± 487.8	$\mathbf{4.4 \pm 3.8}$
PT, $\sigma = 0.001$	29776.5 ± 4804.7	469.0 ± 21.6	20291.5 ± 435.3	$\mathbf{8.9 \pm 1.5}$
PT, $\sigma = 0.01$	30493.0 ± 1470.4	480.0 ± 42.0	20437.0 ± 713.2	$\mathbf{20.6 \pm 3.75}$

ODIN can learn reliable parameters, improving the current state of the art in terms of accuracy and run time.

Accuracy In Figure 4 we compare the trajectory RMSE for the three benchmark systems. While the total tRMSE is an effective indicator for the overall performance, we also include the state-wise tRMSE in the supplementary material. Unfortunately, AGM was unstable on FitzHugh-Nagumo despite serious hyper-prior tuning efforts on our side. We thus do not have any results for this case. To help visualizing the raw numbers obtained by the tRMSE, we also report in Figure 5 the trajectories obtained by numerically integrating the inferred parameters. While here we report only one state for the high noise case of Protein Transduction, a full set of plots can be found in the supplement.

Run time In Table 1, we list the median training times (in seconds) and the corresponding standard deviation of all algorithms on the three parameter inference benchmark systems. It is evident (and not unexpected) that the optimization-based algorithms ODIN and RKG3 are orders of magnitude faster than the MCMC-based FGPGM and AGM. Furthermore, the need for cross-validation schemes in RKG3 seems to increase its run time roughly by an order of magnitude when compared to ODIN.

Identifiability While both LV as well as FHN models are relatively simple, Protein Transduction (PT) still represents a considerable challenge. Amongst others, both Dondelinger et al. (2013) and Wenk et al. (2018) claim that the two parameters θ_5 and θ_6 are only weakly identifiable. However, a quick experiment with different numerical values for those ODE parameters shows that they are actually identifiable. Indeed, neither RKG3 and ODIN seem to suffer from identifiability problems. For ODIN, this can be attributed to two key differences, the inference scheme and the flexible γ. Both AGM and FGPGM ultimately return the posterior mean of the parameter marginals. In Figure 7, we show example marginals for fixed γ. While these distributions are Gaussian-shaped for Lotka-Volterra, they are much wilder for PT. If we were to keep the γ fixed, ODIN would converge to an optimum instead of an expectation, which might be more appropriate in a multi-modal setting. However, ODIN does not keep γ fixed. Instead, its γ evolves during optimization according to the quality of the current parameters estimation, leading to an overall smoother inference. Consequently, the final parameter estimates are significantly more accurate (see Figure 6). For AGM, while the ratio between θ_5 and θ_6 is fairly stable and reasonably not far from the correct number, the absolute parameter values have median magnitudes of roughly 10^{12}: thus they do not appear in this figure.

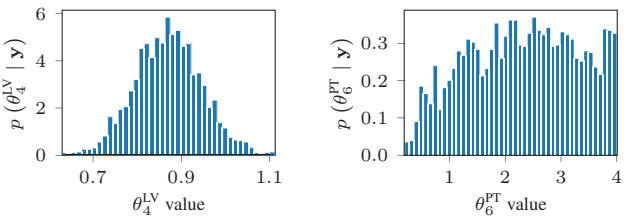


Figure 7: Parameter marginal distributions of θ_4 of Lotka-Volterra and θ_6 of Protein Transduction for one sample rollout with fixed γ. While the LV marginal is nicely Gaussian, the PT marginal is much wilder.

Robustness Besides accurate parameter estimates, ODIN also exhibits more contained variance, especially compared to AGM and RKG3. This is a direct consequence of the underlying GP structure, which enables efficient and stable calculation of all parameters. Furthermore, a flexible γ seems to smooth out the optimization surface, avoiding the rugged landscapes reported by Dondelinger et al. (2013).

Table 2: Median and standard deviation of γ for different model misspecifications of the Lotka-Volterra system and 100 independent noise realizations.

	$\mathcal{M}_{1,1}$	$\mathcal{M}_{0,1}$	$\mathcal{M}_{1,0}$	$\mathcal{M}_{0,0}$
γ_1	$10^{-6} \pm 0.00$	3.01 ± 0.23	$10^{-6} \pm 0.00$	3.03 ± 0.24
γ_2	$10^{-6} \pm 0.04$	$10^{-6} \pm 0.00$	1.51 ± 0.31	1.53 ± 0.35

Priors While in a Bayesian inference setting it is common to introduce a prior over $\boldsymbol{\theta}$, our graphical model in Figure 2 does not treat $\boldsymbol{\theta}$ as a random variable. In a practical setting, we might not even know the parametric form of the ODEs: it thus seems quite difficult to justify the use of a prior. However, it should be noted that our framework can easily accommodate any prior without major modifications. An additional factor $p(\boldsymbol{\theta})$ in Equation (10) directly leads to an additional summand $-\log(p(\boldsymbol{\theta}))$ in Equation (16). From a frequentist perspective, this could be interpreted as an additional regularizer, similarly to LASSO or ridge regression. Since all other summands in Equation (16) grow linearly with the amount of observations N and the prior contribution stays constant, the regularization term would eventually have minor influence in an asymptotic setting.

Model Selection

In practice, domain experts might not be able to provide one single true model. Instead, they might indicate a set of plausible models that they would like to test against the observed data. In this section we investigate this problem, known as model selection. For empirical evaluation, we use the Lotka-Volterra system as ground truth to simulate our empirical data. We then create four different candidate models via the following two additional ODEs

$$\dot{x}_1(t) = \quad \theta_1 x_1^2(t) + \theta_2 x_2(t), \tag{20}$$
$$\dot{x}_2(t) = -\theta_3 x_2(t). \tag{21}$$

Each model is indexed as $\mathcal{M}_{i,j}$, where $i, j \in \{0, 1\}$. Here, $i = 0$ indicates that the wrong equation (i.e. 20) is used to model the dynamics of the first state, while if $i = 1$ we provide the true parametric form in that specific candidate model. In a similar fashion, $j = 0$ indicates that the wrong equation (i.e. 21) is used to model the dynamics of the second state, otherwise $j = 1$. ODIN is run independently for each $\mathcal{M}_{i,j}$. Besides state and parameter estimates, we thus obtain final values for γ, which are presented in Table 2. For numerical stability, γ was lower bounded to 10^{-6} in all experiments. For the correct model $\mathcal{M}_{1,1}$, γ settles at this lower bound, while it converges to a much larger value in case a wrong model is used. This justifies the intuitive interpretation of γ as a mean to account for model mismatch between the GP regressor and the ODE model. This last result proves that γ is indeed an efficient tool for identifying true parametric forms. Interestingly, this also works dimension-wise for the mixed models $\mathcal{M}_{0,1}$ and $\mathcal{M}_{1,0}$, even though the states x_1 and x_2 are coupled via wrong ODEs. This can be explained by the GP regressor prioritizing states $\mathbf{x}$ close to the observations $\mathbf{y}$. Indeed, while incorrect ODEs might deteriorate the accuracy of the state estimates with wrong regularization, their detrimental effects are limited by the observation-dependent partial objective, effectively decoupling the model mismatch across dimensions.

Linear Scaling in State Dimension

A key feature of gradient matching algorithms is the linear scaling in the state dimension K. Following Gorbach, Bauer, and Buhmann (2017), we demonstrate this for ODIN by using the Lorenz '96 system with $\theta = 8$, using 50 observations equally spaced over $t = [0, 5]$. The results are shown in Figure 8, including a linear regressor fitted to the means with least squares.

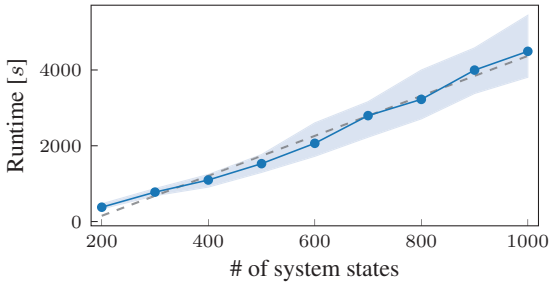


Figure 8: Run time for parameter inference on Lorenz '96 for different state dimension, with a (dashed) linear regressor fitted to the data. For each system size, we report the mean (dots) +- one standard deviation (shaded area) over 100 independent noise realizations.

Discussion

Parametric ODE systems are at the backbone of many practical applications, settings where Gaussian processes and kernel regression have shown to be efficient inference tools. In this paper, we demonstrate how to combine the advantages of both approaches by using theoretical insights to extend standard GP regression. The resulting algorithm, ODIN, significantly improves the current state of the art in terms of accuracy and runtime for parameter inference tasks and provides an appealing framework for model selection. Unlike other methods, ODIN does not require hyperparameter tuning and represents an out-of-the-box applicable tool for parameter inference and model selection for parametric ODE models.

Acknowledgments. This research was supported by the Max Planck ETH Center for Learning Systems. GA acknowledges funding from Google DeepMind and University of Oxford. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme grant agreement No 815943.

References

Abbati, G.; Wenk, P.; Osborne, M. A.; Krause, A.; Schölkopf, B.; and Bauer, S. 2019. Ares and mars adversarial and mmd-minimizing regression for sdes. In *International Conference on Machine Learning*, 1–10.

Barber, D., and Wang, Y. 2014. Gaussian processes for bayesian estimation in ordinary differential equations. In *International Conference on Machine Learning*, 1485–1493.

Bard, Y. 1974. Nonlinear parameter estimation.

Benson, M. 1979. Parameter fitting in dynamic models. *Ecological Modelling* 6(2):97–115.

Calderhead, B.; Girolami, M.; and Lawrence, N. D. 2009. Accelerating bayesian inference over nonlinear differential equations with gaussian processes. In *Advances in neural information processing systems*, 217–224.

Chen, S.; Shojaie, A.; and Witten, D. M. 2017. Network reconstruction from high-dimensional ordinary differential equations. *Journal of the American Statistical Association* 112(520):1697–1707.

Dondelinger, F.; Husmeier, D.; Rogers, S.; and Filippone, M. 2013. Ode parameter inference using adaptive gradient matching with gaussian processes. In *Artificial Intelligence and Statistics*, 216–228.

FitzHugh, R. 1961. Impulses and physiological states in theoretical models of nerve membrane. *Biophysical journal* 1(6):445–466.

González, J.; Vujačić, I.; and Wit, E. 2014. Reproducing kernel hilbert space based estimation of systems of ordinary differential equations. *Pattern Recognition Letters* 45:26–32.

Gorbach, N. S.; Bauer, S.; and Buhmann, J. M. 2017. Scalable variational inference for dynamical systems. In *Advances in Neural Information Processing Systems*, 4806–4815.

Hartikainen, J., and Särkkä, S. 2010. Kalman filtering and smoothing solutions to temporal gaussian process regression models. In *Machine Learning for Signal Processing (MLSP), 2010 IEEE International Workshop on*, 379–384. IEEE.

Hass, H.; Masson, K.; Wohlgemuth, S.; Paragas, V.; Allen, J. E.; Sevecka, M.; Pace, E.; Timmer, J.; Stelling, J.; MacBeath, G.; et al. 2017. Predicting ligand-dependent tumors from multi-dimensional signaling features. *NPJ systems biology and applications* 3(1):27.

Heinonen, M.; Yildiz, C.; Mannerström, H.; Intosalmi, J.; and Lähdesmäki, H. 2018. Learning unknown ode models with gaussian processes. *arXiv preprint arXiv:1803.04303*.

Lorenzi, M., and Filippone, M. 2018. Constraining the dynamics of deep probabilistic models. *arXiv preprint arXiv:1802.05680*.

Lotka, A. J. 1932. The growth of mixed populations: two species competing for a common food supply. *Journal of the Washington Academy of Sciences* 22(16/17):461–469.

Macdonald, B.; Higham, C.; and Husmeier, D. 2015. Controversy in mechanistic modelling with gaussian processes. In *International Conference on Machine Learning*, 1539–1547.

Nagumo, J.; Arimoto, S.; and Yoshizawa, S. 1962. An active pulse transmission line simulating nerve axon. *Proceedings of the IRE* 50(10):2061–2070.

Niu, M.; Rogers, S.; Filippone, M.; and Husmeier, D. 2016. Fast inference in nonlinear dynamical systems using gradient matching. In *Journal of Machine Learning Research: Workshop and Conference Proceedings*, volume 48, 1699–1707. Journal of Machine Learning Research.

Rasmussen, C., and Williams, C. 2006. *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. Cambridge, MA, USA: MIT Press.

Ryder, T.; Golightly, A.; McGough, A. S.; and Prangle, D. 2018. Black-box variational inference for stochastic differential equations. *arXiv preprint arXiv:1802.03335*.

Schölkopf, B.; Herbrich, R.; and Smola, A. J. 2001. A generalized representer theorem. In *International conference on computational learning theory*, 416–426. Springer.

Solak, E.; Murray-Smith, R.; Leithead, W. E.; Leith, D. J.; and Rasmussen, C. E. 2003. Derivative observations in gaussian process models of dynamic systems. In *Advances in neural information processing systems*, 1057–1064.

Stephan, K. E.; Kasper, L.; Harrison, L. M.; Daunizeau, J.; den Ouden, H. E.; Breakspear, M.; and Friston, K. J. 2008. Nonlinear dynamic causal models for fmri. *Neuroimage* 42(2):649–662.

Varah, J. M. 1982. A spline least squares method for numerical parameter estimation in differential equations. *SIAM Journal on Scientific and Statistical Computing* 3(1):28–46.

Vyshemirsky, V., and Girolami, M. A. 2007. Bayesian ranking of biochemical system models. *Bioinformatics* 24(6):833–839.

Wenk, P.; Gotovos, A.; Bauer, S.; Gorbach, N.; Krause, A.; and Buhmann, J. M. 2018. Fast gaussian process based gradient matching for parameter identification in systems of nonlinear odes. *arXiv preprint arXiv:1804.04378*.

Wu, H.; Lu, T.; Xue, H.; and Liang, H. 2014. Sparse additive ordinary differential equations for dynamic gene regulatory network modeling. *Journal of the American Statistical Association* 109(506):700–716.