Fast Inference in Nonlinear Dynamical Systems using Gradient Matching

Mu Niu MU.NIU@GLASGOW.AC.UK
School of Mathematics and Statistics, University of Glasgow, UK

Simon Rogers SIMON.ROGERS@GLASGOW.AC.UK
Department of Computing Science, University of Glasgow,UK

Maurizio Filippone MAURIZIO.FILIPPONE@EURECOM.FR
Eurecom, France

Dirk Husmeier DIRK.HUSMEIER@GLASGOW.AC.UK
School of Mathematics and Statistics, University of Glasgow, UK

Abstract

Parameter inference in mechanistic models of coupled differential equations is a topical problem. We propose a new method based on kernel ridge regression and gradient matching, and an objective function that simultaneously encourages goodness of fit and penalises inconsistencies with the differential equations. Fast minimisation is achieved by exploiting partial convexity inherent in this function, and setting up an iterative algorithm in the vein of the EM algorithm. An evaluation of the proposed method on various benchmark data suggests that it compares favourably with state-of-the-art alternatives.

1. INTRODUCTION

Many processes in science and engineering can be described by dynamical systems based on nonlinear ordinary differential equations (ODEs). Often ODE parameters are unknown and not directly measurable. Since nonlinear ODEs typically have no closed form solution, standard iterative inference procedures require a computationally expensive numerical integration of the ODEs every time the parameters are adapted, which in practice restricts statistical inference to very small systems. To overcome this computational bottleneck, approximate methods based on gradient matching have recently gained much attention. The idea is to circumvent the numerical integration step by using a surrogate cost function that quantifies the discrepancy between the derivative obtained from a smooth interpolant to the data and the derivatives predicted by the ODEs. Various methods have been proposed in the literature, based on P-splines (Ramsay et al., 2007; Liang and Wu, 2008), parallel tempering (Campbell and Steele, 2012), Gaussian processes (Dondelinger et al. (2013), Calderhead et al. (2009), Barber and Wang (2014)), and reproducing kernel Hilbert spaces (RKHS, see González et al. (2013; 2014)). While the application of Gaussian processes in this context has recently been subject to some controversy (Macdonald et al., 2015), the RKHS approach appears to have achieved very promising results (González et al., 2013; 2014) and provides the motivation for the present study.

Consider n arbitrary time points $t_1 < t_2 < \ldots < t_n$ and a set of noisy observations $y_s(t_i)$ of a set of unknown state variables $x_s(t_i)$, $i \in \{1, 2, \ldots, n\}$, $s \in \{1, 2, \ldots, r\}$. The variable $x_s(t)$ represents the value of state variable s at time t, $\boldsymbol{x}(t)$ is an r-dimensional column vector of the values of all state variables at time t, $\boldsymbol{x}_s$ is an n-dimensional row vector of the values of state variable s at all time points, and $\boldsymbol{X} = (\boldsymbol{x}(t_1), \ldots, \boldsymbol{x}(t_n)) = (\boldsymbol{x}_1^\mathsf{T}, \ldots, \boldsymbol{x}_r^\mathsf{T})^\mathsf{T}$ is the matrix of all r state variables at all n time points. We use the same notational convention for the noisy observations $y_s(t)$ with all observations combined into a matrix $\boldsymbol{Y}$. The dynamics of the system composed of the r interacting states x_s, $1 \leq s \leq r$, are governed by coupled non-linear ODEs:

$$\dot{\boldsymbol{x}} = \frac{\partial \boldsymbol{x}}{\partial t} = \boldsymbol{f}\left(\boldsymbol{x}(t), \boldsymbol{\theta}\right), \tag{1}$$

with parameters $\boldsymbol{\theta}$ determining the kinetics of the interactions, and fixed initial values $\boldsymbol{x}(t_1)$ (which, if unknown, can be integrated into $\boldsymbol{\theta}$). We observe or measure the states subject to iid additive Gaussian noise $\boldsymbol{\epsilon}_s \sim N(0, \sigma^2 \boldsymbol{I})$:

$$\boldsymbol{y}_s = \boldsymbol{x}_s + \boldsymbol{\epsilon}_s \tag{2}$$

and the objective of inference is to learn $\boldsymbol{\theta}$ from these noisy measurements or observations. The remainder of the paper is structured as follows: The state-of-the-art in ODE

Proceedings of the 33^{rd} International Conference on Machine Learning, New York, NY, USA, 2016. JMLR: W&CP volume 48.

parameter estimation within the RKHS framework is reviewed in Section 2. Our three-step gradient matching approach based on the RKHS framework is presented in Section 3. Two ODE benchmark models are used for comparative method evaluation in Section 4. We conclude in Section 5 with a discussion of our results.

2. BACKGROUND

A Hilbert space $\mathcal{H}$ is a space of functions g defined over a set $\mathbb{D} \subset \mathbb{R}^m$. $\mathcal{H}$ is said to be a Reproducing Kernel Hilbert Space if and only if there exists a function $k(\cdot,\cdot) : \mathbb{D} \times \mathbb{D} \to \mathbb{R}$ such that for all $t \in \mathbb{D}$ the inner product $< g(\cdot), k(t,\cdot) >$ is equal to $g(t)$ and the kernel function $k(t,\cdot)$ is in $\mathcal{H}$ (Aronszajn, 1950). When working with an RKHS approach for function estimation, functions are expressed in the following form

$$x(t) = \sum_{i=1}^{n} b_i k(t, t_i) \tag{3}$$

with $b_i \in \mathbb{R}$ and $t_i \in \mathbb{D}$. Many possible kernel functions are available including the squared exponential or Radial Basis Function (RBF) kernel, the spline kernel, and the multilayer perceptron (MLP) kernel, to name a few.

The RKHS approach has been previously employed for ODE parameter estimation as follows. Consider a dynamical systems with interacting states denoted by $x_s : \mathbb{D} \to \mathbb{R}$. In equation 2 we interpret $\boldsymbol{x}_s$ as the target regression function to be estimated from the observed data $\boldsymbol{y}_s$. To estimate $\boldsymbol{x}_s$, one needs to make some smoothness assumptions, and a way to do so is to define a likelihood function penalised with a convex regularisation function acting on $\boldsymbol{x}_s$. Green and Silverman (1993) proposed a differential operator to impose smoothness on $\boldsymbol{x}_s$:

$$l_\lambda\left(\boldsymbol{x}_s|\boldsymbol{Y},\sigma^2\right) = -\frac{1}{2\sigma^2}\left\|\boldsymbol{x}_s-\boldsymbol{y}_s\right\|_{L2}^2 - \frac{\lambda}{2}\left\|P\boldsymbol{x}_s\right\|_{L_2}^2, \tag{4}$$

where $\|\cdot\|_{L2}^2$ is the L_2 norm and the P operator is required to be linear. Recent successful RKHS-based approaches to estimating ODE parameters build on this formulation (González et al., 2013; 2014). In their approach, the ODEs are reformulated as a summation of a linear and a nonlinear part. The linear part can be combined with the differential operator to form a linear operator P. The nonlinear part of the ODEs is linearised by feeding a spline interpolation, $\tilde{\boldsymbol{x}}(t)$, of the state vector into the nonlinear part, which we denote by $f_n(\tilde{\boldsymbol{x}}(t),\boldsymbol{\theta})$, giving:

$$\dot{x}_s = f_n(\boldsymbol{x}(t),\boldsymbol{\theta}) - \beta x_s(t) \tag{5}$$

$$\Rightarrow \quad \left(\frac{d}{dt}+\beta\right)x_s(t) = f_n(\tilde{\boldsymbol{x}}(t),\boldsymbol{\theta}) \tag{6}$$

$$\Rightarrow \quad Px_s(t) = f_n(\tilde{\boldsymbol{x}}(t),\boldsymbol{\theta}) \tag{7}$$

Defining $\tilde{\boldsymbol{X}}$ as the matrix of the spline interpolation for all states at all time points, equation (4) becomes:

$$l_\lambda\left(\boldsymbol{x}_s|\boldsymbol{Y},\sigma^2\right) = \frac{-1}{2\sigma^2}\left\|\boldsymbol{x}_s-\boldsymbol{y}_s\right\|_{L2}^2 - \frac{\lambda}{2}\left\|P\boldsymbol{x}_s - f_n(\tilde{\boldsymbol{X}},\boldsymbol{\theta})\right\|_{L_2}^2.$$

The operator P can be connected with a kernel in a Hilbert space as $\mathcal{K} = (P^*P)^{-1}$ (Green and Silverman, 1993). However, this poses a computational challenge, as the analytical solution of this inversion is not available in closed form and calls for the need of an approximation. González et al. (2014) approximate the differential operator by a differencing operator using a finite element method, making the linear operator P become $\boldsymbol{D}+\beta\boldsymbol{I}$, where $\boldsymbol{D}$ is the difference matrix (see González et al. (2014) for the explicit expression). The inverse of the kernel Gram matrix can then be approximated as:

$$\boldsymbol{\mathcal{K}}_G^{-1} = \left(\boldsymbol{D}+\beta\boldsymbol{I}\right)^T\left(\boldsymbol{D}+\beta\boldsymbol{I}\right) \tag{8}$$

and eq. (4) can be rewritten as (see González et al. (2014)):

$$l_\lambda\left(\boldsymbol{x}_s|\boldsymbol{Y},\sigma^2,\boldsymbol{\theta}\right) = -\frac{\left\|\tilde{\boldsymbol{y}}_s-\boldsymbol{\mathcal{K}}_G\boldsymbol{\alpha}\right\|^2}{2\sigma^2} - \frac{\lambda}{2}\boldsymbol{\alpha}^T\boldsymbol{\mathcal{K}}_G\boldsymbol{\alpha} \tag{9}$$

$$\boldsymbol{\alpha} = \left(\boldsymbol{\mathcal{K}}_G+\lambda\sigma^2\mathbf{I}\right)^{-1}\tilde{\boldsymbol{y}}_s \tag{10}$$

$$\tilde{\boldsymbol{y}}_s = \boldsymbol{y}_s - P^{-1}f_n(\tilde{\boldsymbol{X}},\boldsymbol{\theta}) \tag{11}$$

The ODE parameters are then obtained by maximisation of eq. (9) with respect to $\boldsymbol{\theta}$, which is a regularisation problem in RKHS (Berlinet and Thomas-Agnan, 2011). The authors found that their method outperformed the alternative ODE parameter estimation techniques of Ramsay et al. (2007) and Khanin et al. (2007).

However, there are two drawbacks of this approach. Firstly, while the approximation of the derivative operator by a difference operator is reliable for time series sampled at high frequencies, it tends to perform poorly with sparse and noisy data. Secondly, the linearisation of the nonlinear part introduces inaccuracies as it is based on simple spline interpolation with no influence from the ODEs. In the next section, we propose a new approach, which is also based on the RKHS framework, but avoids these difficulties.

3. PROPOSED METHOD

Like González et al. (2013; 2014), we model the unknown concentrations of the sth component of the dynamical system in eq.(1) at time t with a linear combination of kernels $k(.,.)$ from some function family $\mathcal{F}$:

$$g_s(t;\boldsymbol{b}_s) = \sum_{j=1}^{n} b_{sj}k(t,t_j) \tag{12}$$

We denote by $\boldsymbol{b}_s$ the vector of kernel regression coefficients b_{sk} and define $\boldsymbol{B} = (\boldsymbol{b}_1^\mathsf{T},...,\boldsymbol{b}_r^\mathsf{T})$, where r denotes the total

number of components in the system. We estimate $\boldsymbol{B}$ along with $\boldsymbol{\theta}$ by minimisation of the following objective function:

$$E(\boldsymbol{\theta}, \boldsymbol{B}) = \sum_{s=1}^{r} \left(\sum_{i=1}^{n} \Big[g_s(t_i; \boldsymbol{b}_s) - y_{si} \Big]^2 \right) + \rho \sum_{s=1}^{r} \left(\sum_{i=1}^{n} \Big[\dot{g}_s(t_i; \boldsymbol{b}_s) - f_s(\boldsymbol{g}(t_i, \boldsymbol{B}), \boldsymbol{\theta}) \Big]^2 \right) \quad (13)$$

where $\boldsymbol{g}(t_i, \boldsymbol{B}) = (g_1(t_i; \boldsymbol{b}_1), \ldots, g_r(t_i; \boldsymbol{b}_r))^{\mathsf{T}}$, and $\rho \geq 0$ is a regularisation parameter that can be estimated efficiently (by parallelisation) with 10-fold cross-validation. The first term penalises deviations of the interpolant $g_s(t_i; \boldsymbol{b}_s)$ from the data y_{si}. The second term is a gradient matching term that penalises the difference between the gradient obtained from the interpolant,

$$\dot{g}_s(t; \boldsymbol{b}_s) = \sum_{i=1}^{n} b_{si} \frac{dk(t, t_i)}{dt} = \sum_{i=1}^{n} b_{si} \dot{k}(t, t_i) \quad (14)$$

and the gradient predicted from the ODEs, $f_s(\boldsymbol{g}(t_i, \boldsymbol{B}), \boldsymbol{\theta})$. Minimising $E(\boldsymbol{\theta}, \boldsymbol{B})$ with respect to $\boldsymbol{B}$ for given $\boldsymbol{\theta}$ is a regularised regression problem that aims to minimise the sum-of-squares error subject to penalising interpolants that are not consistent with the ODEs. Minimising $E(\boldsymbol{\theta}, \boldsymbol{B})$ with respect to $\boldsymbol{\theta}$ for given $\boldsymbol{B}$ estimates the ODE parameters via gradient matching. The inference problem

$$\{\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{B}}\} = \text{argmin}_{\boldsymbol{\theta}, \boldsymbol{B}} E(\boldsymbol{\theta}, \boldsymbol{B})$$

faces two practical problems. Firstly, $E(\boldsymbol{\theta}, \boldsymbol{B})$ is usually multimodal. This calls for a 'good' initialisation of $\{\boldsymbol{\theta}, \boldsymbol{B}\}$ that is 'close' to the global minimum. Secondly, $E(\boldsymbol{\theta}, \boldsymbol{B})$ is non-convex in both $\boldsymbol{\theta}$ and $\boldsymbol{B}$. This leaves us with a computationally expensive optimisation problem that calls for numerical acceleration. In addition, the kernels $k_s(t_k, t_i)$ typically depend on some hyperparameters $\boldsymbol{\varphi}_s$ (e.g. the length scale of an RBF kernel), which need to be set in advance according to a separate optimality criterion. We discuss these issues in the remainder of this section. In what follows, $\boldsymbol{y}_s$ represents the vector of observations for the sth state, and $\boldsymbol{K}_s$ is the Gram matrix with entries $k_s(t_k, t_i)$.

Step 1 - Initialisation of regression parameters and optimisation of kernel hyperparameters. Following standard kernel ridge regression, the interpolants $g_s(t)$ from eq.(12) are obtained by minimising the following regularised loss function:

$$\mathcal{L}(\boldsymbol{b}_s, \boldsymbol{\varphi}_s; \lambda_s) = \sum_{i=1}^{n} \Big(g_s(t_i; \boldsymbol{b}_s) - y_{si} \Big)^2 + ||\boldsymbol{g}_s||^2 \quad (15)$$

where the dependence on $\boldsymbol{\varphi}_s$ is via k_s (which has not been made explicit in the notation), and the regularisation term $||\boldsymbol{g}_s||^2$ is the squared norm in $\mathcal{H}_s$, $||\boldsymbol{g}_s||^2 = \lambda_s \boldsymbol{b}_s^T \boldsymbol{K}_s \boldsymbol{b}_s$, which contains a regularisation parameter $\lambda_s \geq 0$. The minimisation of $\mathcal{L}(\boldsymbol{b}_s, \boldsymbol{\varphi}_s; \lambda_s)$ with respect to $\boldsymbol{b}_s$ for given $\boldsymbol{\varphi}_s$ and λ_s is a convex optimisation problem with solution

$$\boldsymbol{b}_s = (\boldsymbol{K}_s + \lambda_s \boldsymbol{I})^{-1} \boldsymbol{y}_s \quad (16)$$

Given λ_s, the kernel hyper-parameters $\boldsymbol{\varphi}_s$ are optimised independently with a standard optimisation routine, like trust region or quasi-Newton. The regularisation parameters λ_s are estimated using 10-fold cross validation (parallelised!).

Step 2 - Initialisation of ODE parameters using gradient matching. Setting $\boldsymbol{B}$ fixed at the values obtained from Step 1, the ODE parameters $\boldsymbol{\theta}$ are obtimised by minimising the objective function E of eq. (13) using a standard optimisation routine (e.g. trust region or quasi Newton).

The combination of Steps 1 and 2 provides a straightforward method for ODE parameter inference, which we henceforth refer to as the RKG2 method: smooth regression with standard regularisation to prevent overfitting (Step 1) followed by ODE parameter estimation via gradient matching (Step 2). What is missing is a regularising influence of the ODEs back on the interpolation. This is effected by minimising the objective function $E(\boldsymbol{\theta}, \boldsymbol{B})$ of eq. (13) in the following step.

Step 3 - Minimisation of the combined objective function with convergence acceleration. The subsequent minimisation of $E(\boldsymbol{\theta}, \boldsymbol{B})$ with respect to both arguments is a complex non-convex optimisation problem. However, when fixing $\boldsymbol{B}$ in the argument of $f_s(.)$, the objective function is convex in the remaining parameters $\boldsymbol{B}$, due to the linearity in eq. (12). This convexity can be exploited with a modified optimisation algorithm, which bears a certain resemblance to the EM algorithm. First, we define the following modified objective function:

$$\tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}, \tilde{\boldsymbol{B}}) = \sum_{s=1}^{r} \left(\sum_{i=1}^{n} \Big[g_s(t_i; \boldsymbol{b}_s) - y_{si} \Big]^2 \right) + \rho \sum_{s=1}^{r} \left(\sum_{i=1}^{n} \Big[\dot{g}_s(t_i; \boldsymbol{b}_s) - f_s(\boldsymbol{g}(t_i, \tilde{\boldsymbol{B}}), \boldsymbol{\theta}) \Big]^2 \right) \quad (17)$$

Note that $\tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}, \boldsymbol{B}) = E(\boldsymbol{\theta}, \boldsymbol{B})$. We now carry out the following iteration until reaching a zero-gradient point:

1. Given $\boldsymbol{B}$ and $\boldsymbol{\theta}$, minimize $\tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}^*, \boldsymbol{B})$ with respect to $\boldsymbol{B}^*$, i.e. find $\boldsymbol{B}_{new} = \text{argmin}_{\boldsymbol{B}^*} \tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}^*, \boldsymbol{B})$
2. Set $\boldsymbol{B} = \boldsymbol{B}_{new}$ and minimise $\tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}_{new}, \boldsymbol{B}_{new})$ wrt $\boldsymbol{\theta}$, i.e. find $\boldsymbol{\theta}_{new} = \text{argmin}_{\theta} \tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}_{new}, \boldsymbol{B}_{new})$

Theorem. Let $\mathcal{G}$ denote the set of functions defined by equation (12). Assume $\mathcal{G}$ is contained in the solution space

of the ODEs in the sense that $\forall \boldsymbol{g} \in \mathcal{G}\ \exists \boldsymbol{\theta} \in \mathbb{R}: \ \dot{\boldsymbol{g}} = f(\boldsymbol{g}, \boldsymbol{\theta})$. Then each parameter adaptation step of the algorithm described above, $(\boldsymbol{B}, \boldsymbol{\theta}) \rightarrow (\boldsymbol{B}_{new}, \boldsymbol{\theta}_{new})$, implies that $E(\boldsymbol{\theta}_{new}, \boldsymbol{B}_{new}) \leq E(\boldsymbol{\theta}, \boldsymbol{B})$, and the iteration converges to a zero gradient point of $E(\boldsymbol{\theta}, \boldsymbol{B})$.

Proof. The first step of the algorithm implies that

$$\tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}_{new}, \boldsymbol{B}) \ \leq \ \tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}, \boldsymbol{B}) \ = \ E(\boldsymbol{\theta}, \boldsymbol{B}) \qquad (18)$$

For the second step, note that $\exists \boldsymbol{\theta}^* \in \mathbb{R}$ such that $\dot{\boldsymbol{g}}(t, \boldsymbol{B}_{new}) = f(\boldsymbol{g}(t, \boldsymbol{B}_{new}), \boldsymbol{\theta}^*)$, by assumption of the theorem. This implies that

$$||\dot{\boldsymbol{g}}(t, \boldsymbol{B}_{new}) - f(\boldsymbol{g}(t, \boldsymbol{B}_{new}), \boldsymbol{\theta}^*)||^2 = 0 \ \ \forall t$$

Hence $\tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}_{new}, \boldsymbol{B}) =$

$$\begin{aligned}
&= \sum_{i=1}^{N} \Big\{ ||\boldsymbol{y}(t_i) - \boldsymbol{g}(t_i, \boldsymbol{B}_{new})||^2 + \\
&\rho ||\dot{\mathbf{g}}(t_i, \boldsymbol{B}_{new}) - f[\boldsymbol{g}(t_i, \boldsymbol{B}), \boldsymbol{\theta}]||^2 \Big\} \\
&\geq \sum_{i=1}^{N} \Big\{ ||\boldsymbol{y}(t_i) - \boldsymbol{g}(t_i, \boldsymbol{B}_{new})||^2 \Big\} \ + \ 0 \\
&= \sum_{i=1}^{N} \Big\{ ||\boldsymbol{y}(t_i) - \boldsymbol{g}(t_i, \boldsymbol{B}_{new})||^2 + \\
&\rho ||\dot{\mathbf{g}}(t_i, \boldsymbol{B}_{new}) - f[\boldsymbol{g}(t_i, \boldsymbol{B}_{new}), \boldsymbol{\theta}^*]||^2 \Big\} \\
&= \tilde{E}(\boldsymbol{\theta}^*, \boldsymbol{B}_{new}, \boldsymbol{B}_{new})
\end{aligned}$$

This implies that on completion of the second step of the algorithm, $\boldsymbol{\theta}_{new} = \text{argmin}_{\vartheta} \tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}_{new}, \boldsymbol{B}_{new})$, we have $\tilde{E}(\boldsymbol{\theta}_{new}, \boldsymbol{B}_{new}, \boldsymbol{B}_{new}) = \tilde{E}(\boldsymbol{\theta}^*, \boldsymbol{B}_{new}, \boldsymbol{B}_{new})$, and hence

$$\begin{aligned}
E(\boldsymbol{\theta}_{new}, \boldsymbol{B}_{new}) \ &= \ \tilde{E}(\boldsymbol{\theta}_{new}, \boldsymbol{B}_{new}, \boldsymbol{B}_{new}) \\
&= \ \tilde{E}(\boldsymbol{\theta}^*, \boldsymbol{B}_{new}, \boldsymbol{B}_{new}) \ \leq \ \tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}_{new}, \boldsymbol{B}) \qquad (19)
\end{aligned}$$

Combining equations (18) and (19), we get:

$$E(\boldsymbol{\theta}_{new}, \boldsymbol{B}_{new}) \ \leq \ \tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}_{new}, \boldsymbol{B}) \ \leq \ E(\boldsymbol{\theta}, \boldsymbol{B})$$

which completes the first part of the proof. The iteration is continued until $\nabla_{\boldsymbol{\theta}} \tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}, \boldsymbol{B}) = \nabla_{\boldsymbol{B}} \tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}, \boldsymbol{B}) = 0$. Since $\tilde{E}(\boldsymbol{\theta}, \boldsymbol{B}, \boldsymbol{B}) = E(\boldsymbol{\theta}, \boldsymbol{B})$, this is a zero-gradient point of the original objective function $E(\boldsymbol{\theta}, \boldsymbol{B})$.

Table 1 shows that the proposed algorithm accelerates convergence by two orders of magnitude over the direct minimisation of the objective function of eq. (13) with standard iterative optimisation methods, like trust-region or quasi-Newton minimisation. The improvement stems from the fact that the first step of the proposed iteration is a quadratic optimisation problem, with closed-form solution:

$$\boldsymbol{b}_s = \left(\boldsymbol{K}_s^T \boldsymbol{K}_s + \dot{\boldsymbol{K}}_s^T \dot{\boldsymbol{K}}_s \right)^{-1} \left(\boldsymbol{K}_s \boldsymbol{y}_s(\boldsymbol{t}) + \dot{\boldsymbol{K}}_s \boldsymbol{f}_s(\boldsymbol{t}, \boldsymbol{\theta}, \boldsymbol{B}) \right)$$

Table 1. **Comparison of computational costs.** The table shows the computational costs for a single iteration of two optimisation algorithms compared in our study, and an alternative method discussed in Section 2, using the data generated from eq. (20).

Method	CPU time
Direct minimisation of $E(\boldsymbol{\theta}, \boldsymbol{B})$	$599.8s$
Proposed acceleration algorithm (RKG3)	$6.7s$
RKHS method by Gonzalez et al. (GON)	$4.2s$

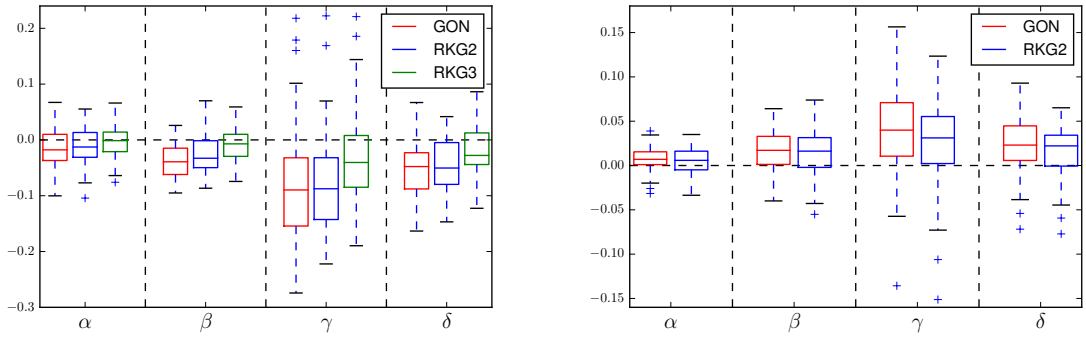
where $\boldsymbol{f}_s(\boldsymbol{t}, \boldsymbol{\theta}, \boldsymbol{B}) = (f_s(t_1, \boldsymbol{\theta}, \boldsymbol{B}), \ldots, f_s(t_n, \boldsymbol{\theta}, \boldsymbol{B}))^{\mathsf{T}}$, $\boldsymbol{K}_s$ is the Gram kernel matrix for the sth state and $\dot{\boldsymbol{K}}_s$ is the corresponding matrix of derivatives with respect to time. In practice, we avoid the potentially numerically unstable matrix inversion by appropriate factorisation and updating the $\boldsymbol{b}_s$ by means of forward and back substitutions. The assumption of containment, on which the proof is based, is quite restrictive, and the objective of our future work is to generalise the proof to more relaxed conditions. Our simulations suggest that the acceleration algorithm is effective without having to satisfy the containment condition. We provide two examples in the following section.

Table 2. **Statistical hypothesis test for the Lotka-Volterra data.** The table shows the p-values corresonding to Figures 1 and 2. Standard fonts: no significat difference (p-value > 0.05). Bold fonts: the proposed RKG3 method significantly outperforms the competing method (p-value < 0.05).

Par	GON $\sigma = 0.25$	RKG2 $n = 34$	GON $\sigma = 0.4$	RKG2 $n = 51$
$\boldsymbol{\alpha}$	**3.8e-3**	**2.5e-2**	5e-1	1.7e-1
$\boldsymbol{\beta}$	**2.2e-5**	**5.4e-4**	7e-1	1.4e-1
$\boldsymbol{\gamma}$	**3.2e-6**	**4.8e-4**	**7e-5**	**1.5e-6**
$\boldsymbol{\delta}$	**2.1e-5**	**5.7e-4**	6.1e-2	**6.7e-4**

Table 3. **Method evaluation on the Lotka-Volterra data.** Performance criteria are the root median square error in parameter space (par) and function space (fun; for the functions obtained by inserting the estimated parameters into the ODEs). Values in brackets show the median absolute deviation (MAD). The true parameter values are: $\alpha = 0.2$, $\beta = 0.35$, $\gamma = 0.7$, $\delta = 0.4$.

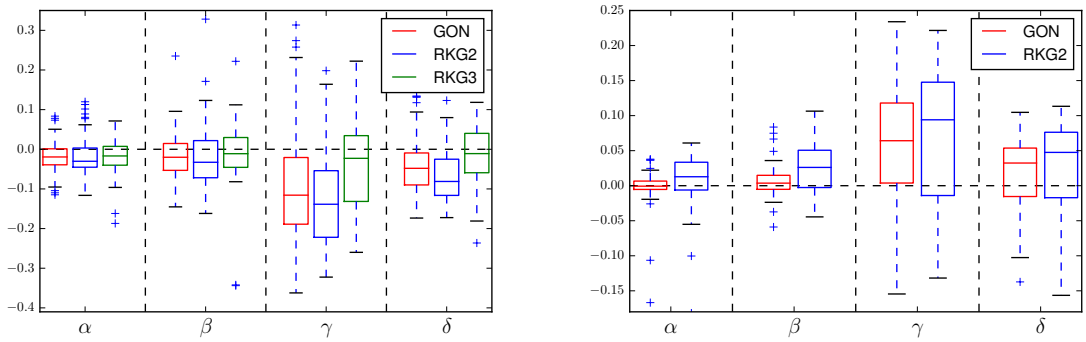
$\boldsymbol{\sigma}$	**n**	**Method**	**par**$_{10^{-2}}$	**fun**$_{10^{-2}}$
0.25	34	$RKG3$	4.6(2.3)	64.4(37.8)
		$RKG2$	6.1(3.2)	77.6(66)
		GON	7.1(3.3)	98.7(55)
0.4	51	$RKG3$	5.4(3.3)	95.8(72.7)
		$RKG2$	10.3(4.7)	151(56.3)
		GON	7.9(5)	129(66)



(a) Parameter estimation

(b) Parameter estimation differences, compared with RKG3

Figure 1. **Method evaluation on the Lotka-Volterra data, lower noise level (10 db).** The figure shows distributions of parameter estimates from 50 data instantiations, generated from the Lotka-Volterra system, with noise level $\sigma = 0.25$ (10db) and sample size $n = 34$. (a) Distributions of the parameter differences (inferred value minus true value); from left to right: GON, RKG2, RKG3. The dashed horizontal line indicates no difference from the true value. (b) Distribution of the absolute differences, $|A - L| - |RKG3 - L|$, where $RKG3$ is the estimate obtained with the proposed RKG3 method, A is the estimate obtained with the alternative, and L is the true value. The dashed horizontal line indicates equal performance. Positive values indicate that RKG3 outperforms the alternative method. A t-test was carried out to test the significance of the indicated trends. The corresponding p-values are shown in Table 2.



(a) Parameter estimation

(b) Parameter estimation differences, compared with RKG3

Figure 2. **Method evaluation on the Lotka-Volterra data, higher noise level (6 db).** The figure corresponds to Figure 1, with the noise standard deviation increased to $\sigma = 0.4$(6 db), and the sample size increased to $n = 51$. For details, see the caption of Figure 1. The p-values from a paired t-test corresponding to the right panel are shown in 2.

4. SIMULATIONS

The objective of our simulation study is to compare the performance of three algorithms: the RKHS-based method proposed by González et al. (2013; 2014) (GON), the method proposed in the previous section (RKG3), and a faster, reduced version of this method, which only carries out the first two steps of the algorithm (RKG2). This is to assess the effectiveness of the full optimisation of the objective function in eq. (13), which effecitvely constitutes a regularisation effect by which the ODEs reshape the interpolant obtained from the smooth regression in Step 1.

We have evaluated the methods on data generated from two ODE systems: the classical Lotka-Volterra system, and a mathematical description of a protein signal transduction pathway. The Lotka-Volterra equations describe the dynamics of ecological systems with predator-prey interactions (Lotka, 1920):

$$\dot{x_1} = \alpha \cdot x_1 - \beta \cdot x_1 \cdot x_2, \; \dot{x_2} = -\gamma \cdot x_2 + \delta \cdot x_1 \cdot x_2$$

where the dot denotes a derivative with respect to time, $\alpha, \beta, \gamma, \delta$ are four parameters to be inferred, and x_1 and x_2 are the states of the model, indicating the numbers of prey and predators, respectively. We numerically solved

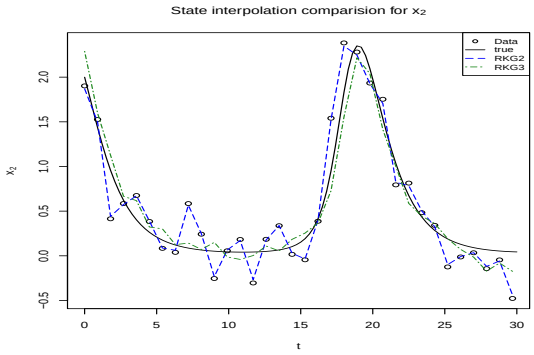


Figure 3. **Demonstration of the regularisation effect of step 3 of the RKG3 method.** The figure shows the true solution (solid line) of the Lotka-Volterra system, noisy observations ($\sigma = 0.25$) of sample size $n = 34$ (circles), the interpolant obtained with the RKG2 method (dashed line), and the interpolant obtained with the RKG3 method (dash-dotted line). It is seen that the RKG2 interpolant shows clear signs of overfitting, and that the RKG3 interpolant shows much better agreement with the true signal. This demonstrates the effectiveness of the regularisation inherent in the third step of the proposed RKG3 method, by which the ODEs act back as a regulariser on the interpolant.

the ODEs for $\alpha = 0.2$, $\beta = 0.35$, $\gamma = 0.7$, $\delta = 0.4$ and initial conditions $x_1(0) = 1$ and $x_2(0) = 2$. We generated 50 independent noisy observation of x_1 and x_2 by adding zero mean iid Gaussian noise with standard deviations $\sigma = 0.25$ (signal-to-noise ratio $SNR = 10db$) and $\sigma = 0.4$ ($SNR = 6db$). We recorded samples of sample sizes $n \in \{34, 51\}$ evenly spaced in the interval of $[0, 30]$.

A model for the interactions of five protein isoforms, S, dS, R, RS, Rpp, in a signal transduction pathway was studied by Vyshemirsky and Girolami (2008), based on mass action and Michaelis-Menten kinetics:

$$\begin{aligned}
[\dot{S}] &= -k_1 \cdot [S] - k_2 \cdot [S] \cdot [R] + k_3 \cdot [RS] \\
[\dot{dS}] &= k_1 \cdot [S] \\
[\dot{R}] &= -k_2 \cdot [S] \cdot [R] + k_3 \cdot [RS] + \frac{k_5 \cdot [Rpp]}{k_6 + [Rpp]} \\
[\dot{RS}] &= k_2 \cdot [S] \cdot [R] - k_3 \cdot [RS] - k_4 \cdot [RS] \\
[\dot{Rpp}] &= k_4 \cdot [RS] - \frac{k_5 \cdot [Rpp]}{k_6 + [Rpp]}
\end{aligned} \tag{20}$$

The square brackets, $[\cdot]$, denote concentrations, and the letters $k_{1:6}$ represent 6 kinetic parameters to be inferred. It turns out that k_5 and k_6 are only weakly identifiable, and we have thus assessed the accuracy of inference based on the ratio $\frac{k_5}{k_6}$. We took the kinetic parameters from Vyshemirsky and Girolami (2008) and generated 50 independent data instantiations of different sample size, $n = 14$ and $n = 28$, and different noise standard deviations: $\sigma = 0.01$ ($SNR = 24db$) and $\sigma = 0.052$ ($SNR = 10db$).

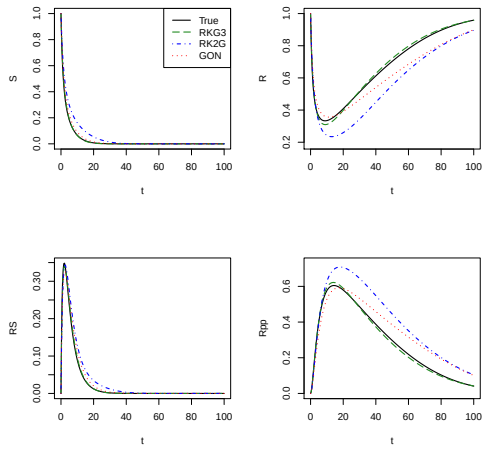


Figure 4. **Method evaluation in function space.** The parameters estimated using the different methods were fed back into the ODEs and the solutions were plotted and compared with the true solution. Black solid line: true solution. Red dotted line: GON. Blue dash-dotted line: RKG2. Green dashed line: RKG3. The inference was based on data obtained from the protein signal transduction pathway model of eq.(20) with sample size $n = 14$ and noise level $\sigma = 0.01$. RKG3 gives the best approximation.

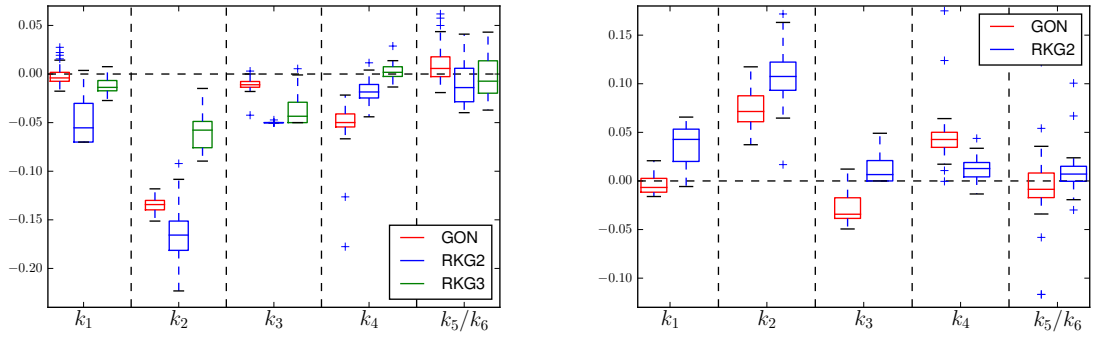
A numerical solution of the Lotka-Volterra system gives stationary oscillations, and we therefore chose a stationary kernel: the RBF kernel with state-specific lengthscale l_s:

$$k_s(t_k, t_i) = \exp\left(-l_s^{-2}(t_k - t_i)^2\right) \tag{21}$$

The protein concentrations obtained from the protein transduction pathway are nonstationary, as seen from Figure 4, and we have therefore chosen a non-stationary kernel: the multi-layer perceptron kernel (MLP), with state-specific parameters w_s and l_s, given by

$$k(t_k, t_i) = \arcsin\left(\frac{wt_k t_i + l}{\sqrt{wt_i^2 + l + 1}\sqrt{wt_k^2 + l + 1}}\right) \tag{22}$$

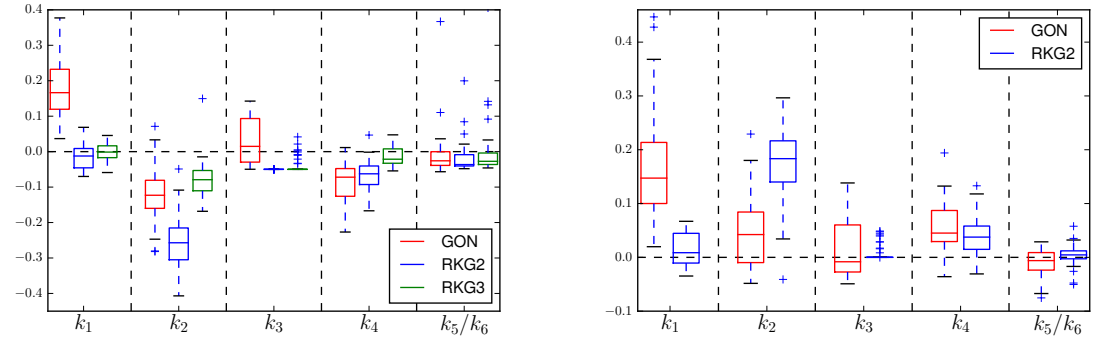
For the GON method, we used the authors' software, for RKG2 and RKG3 we used our own code, which is available upon request. The computational costs are shown in Table 1. Figure 3 illustrates the improvement obtained with the third step of the proposed method (RKG3), showing much better agreement with the true solution than the interpolant obtained with RKG2 and thereby indicating the efficacy of the regularisation effect inherent in the feedback of the ODEs acting back on the interpolant via minimising the objective function in eq. (13). The left panels of Figures 1-2,5-6 show the distributions (over 50 independent data instantiations) of the parameters inferred with the different



(a) Parameter estimation

(b) Parameter estimation differences, compared with RKG3

Figure 5. **Method assessment on the protein pathway data, lower noise level (24 db).** The figure shows the same distributions as in Figure 1, but obtained from 50 data instantiations of the protein signalling pathway of eq.(1). Noise standard deviation: $\sigma = 0.01$(24db). Sample size: $n = 14$. For details of the box plots, see the caption of Figure 1. A paired t-test was carried out, to test the significance of the indicated trends; the p-values are shown in Table 4.



(a) Parameter estimation

(b) Parameter estimation differences, compared with RKG3

Figure 6. **Method assessment on the protein pathway data, higher noise level (10 db).** The figure corresponds to Figure 5, but with the noise increased to $\sigma = 0.052$(10db), and the sample size increased to $n = 28$. See the caption of Figure 5 for details. The p-values from a paired t-test corresponding to the right panel are shown in Table 4.

methods. For a clearer comparison between the methods, we computed the absolute differences between the inferred and true parameters, and then subtracted the score obtained with RKG3 from the scores obtained with the competing methods. A positive value indicates that the parameters obtained with RKG3 are closer to the true parameters. The right panels of Figures 1-2,5-6 confirm that, for most parameters, this is indeed the case. We followed this graphical presentation up with a statistical hypothesis test, shown in Tables 2 and 4, which shows that the improvement obtained with RKG3 is significant in 69% of the tests performed, and that RKG3 is never significantly worse than any of the the competing methods. Tables 3 and 5 show the root median square distance in parameter space and function space, where the latter measure was obtained by reinserting the estimated parameters into the ODEs, numerically solving them, and comparing the results with the true solution. In all cases, we computed the median absolute deviation for uncertainty quantification. Our findings suggest that RKG3 clearly outperforms the alternative methods.

5. DISCUSSION

We have proposed an approach to parameter estimation in ODEs that overcomes the need for computationally expensive numerical integration (RKG3). The approach consists of an iterative three-step procedure. The first step carries out smooth function interpolation within the reproducing kernel Hilbert space framework. The second step estimates the parameters of the ODEs based on the principle of gra-

Table 4. **Statistical hypothesis test for the protein pathway data.** The table shows the p-values corresponding to Figures 5 and 6. Bold fonts: the proposed RKG3 method significantly outperforms the alternative schemes. Standard fonts: the difference is not significant.

Par	GON $\sigma = 0.01$	RKG2 $n = 14$	GON $\sigma = 0.052$	RKG2 $n = 28$
k_1	**3.6e-4**	**6.6e-17**	**2.1e-11**	**2.5e-3**
k_2	**3.2e-31**	**3.4e-31**	**1.1e-4**	**3.9e-20**
k_3	**6.3e-17**	**2e-7**	1.6e-1	**2.5e-3**
k_4	**1.5e-16**	**1.3e-8**	**7.8e-10**	**2e-9**
$\frac{k_5}{k_6}$	2.1e-1	1.6e-1	3.1e-1	2.1e-1

Table 5. **Method evaluation on the protein pathway data.** Performance criteria are the root median square error in parameter space (par) and function space (fun; for the functions obtained by inserting the estimated parameters into the ODEs). The values in brackets show the median absolute deviation (MAD). The true value of the parameters are fixed to $k_1 = 0.07$,$k_2 = 0.6$,$k_3 = 0.05$,k_4=0.3,$k_5 = 0.017$,$k_6 = 0.3$

σ	n	Method	$\text{par}_{10^{-2}}$	$\text{fun}_{10^{-2}}$
0.01	14	RKG3	3.7(1)	5.2(1.6)
		RKG2	8.6(1.4)	22.7(8.4)
		GON	6.5(0.4)	9.1(5.2)
0.052	28	RKG3	5.4(2)	11(6)
		RKG2	12.5(2.8)	18.8(7)
		GON	11(5.3)	37(11)

dient matching. The third step, inherent in the minimisation of the objective function in eq. (13), regularises the interpolant by balancing goodness-of-fit against a discrepancy measure between the estimated derivatives and those predicted from the ODEs. This scheme is naturally iterated until some convergence criterion is met. To reduce the computational costs of the third step, we have exploited the fact that the explicit dependence of the objective function on the regression parameters is quadratic, which suggests an iterative procedure akin to the EM algorithm: solving a convex optimisation problem for fixed parameters within the ODEs (akin to an M-step), then reinserting these parameters into the ODEs (akin to an E-step).

We have evaluated our method on two systems of differential equations: the Lotka-Volterra system, and a mathematical description of a protein signalling pathway. We have compared the performance of our method with the RKHS-based method proposed by González et al. (2013; 2014) (GON). The authors report that their method achieves a similar performance as methods based on a computationally far more expensive explicit solution of the ODEs, and that it outperforms the method proposed in the seminal work of (Ramsay et al., 2007). Hence, GON appears to be representative of the current state of the art. To evaluate the effectiveness of the regularisation step (Step 3 in our algorithm) we also carried out a comparison with a reduced 2-step version of our algorithm (RKG2), in which the regularisation inherent in the minimisation of the objective function in eq. (13) is missing. Our evaluation is based on inspecting the distributions of the absolute difference between the estimated and true parameters over a large number of independent data instantiations (Figures 1-2,5-6), testing the statistical significance of the difference (Tables 2 & 4), and quantifying an average distance metric in parameter as well as in function space (Tables 3 & 5).

Our results indicate that in most cases, the proposed RKG3 algorithm achieves a significant improvement over the two alternative methods. For those parameters where no improvement could be found, the differences were not significant, indicating that RKG3 is always at least as good as the other two methods. The effectiveness of the regularisation scheme inherent in the minimisation of the objective function in eq. (13) is illustrated in Figure 3, and the performance improvement in function space is illustrated in Figure 4. The improvement of the proposed method over GON stems from the fact that it does not need to approximate time derivatives by difference quotients, and that the estimation of all component concentration profiles takes both data mismatch and consistency with the ODEs into consideration, by virtue of eq. (13). We emphasise that a naive optimisation of the objective function in eq. (13) would substantially increase the computational complexity over that of GON. However, by exploiting partial convexity inherent in the objective function, we have reduced the computational complexity by about two orders of magnitude, bringing it into the same range as GON.

We emphasise that the focus of our work has been on fast inference. We have therefore avoided Bayesian sampling methods and tried to keep the methodology as closely as possible within the realm of convex optimisation. We note that substantial recent research efforts have been invested in ODE parameter inference with Bayesian state space models (e.g. Baker et al. (2011)) and emulators (e.g.Wilkinson (2014)). These methods clearly have the potential to achieve a high degree of accuracy, but they also critically hinge on a 'good' initialisation. This becomes particularly evident for statistical emulation, where the initial space-filling design in a high-dimensional parameter space is computationally onerous.To turn emulation into a viable tool, the initial parameter space needs to be confined to a small subdomain deemed plausible a priori. It is here that the method proposed in the present article will provide a powerful complementary tool, by enabling fast ODE parameter estimates for guidance on the initial design.

Acknowledgements

This work was supported by EPSRC (EP/L020319/1).

References

Nachman Aronszajn. Theory of reproducing kernels. *Transactions of the American mathematical society*, pages 337–404, 1950.

Syed Murtuza Baker, C Hart Poskar, and Bjorn H Junker. Unscented kalman filter with parameter identifiability analysis for the estimation of multiple parameters in kinetic models. *EURASIP Journal on Bioinformatics and Systems Biology*, 2011.

David Barber and Yali Wang. Gaussian processes for Bayesian estimation in ordinary differential equations. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1485–1493, 2014.

Alain Berlinet and Christine Thomas-Agnan. *Reproducing kernel Hilbert spaces in probability and statistics*. Springer Science & Business Media, 2011.

Ben Calderhead, Mark Girolami, and Neil D Lawrence. Accelerating Bayesian inference over nonlinear differential equations with Gaussian processes. In *Advances in Neural Information Processing Systems*, pages 217–224, 2009.

David Campbell and Russell J Steele. Smooth functional tempering for nonlinear differential equation models. *Statistics and Computing*, 22(2):429–443, 2012.

Frank Dondelinger, Maurizio Filippone, Simon Rogers, and Dirk Husmeier. ODE parameter inference using adaptive gradient matching with Gaussian processes. In *Sixteenth International Conference on Artificial Intelligence and Statistics*, 2013.

Javier González, Ivan Vujačić, and Ernst Wit. Inferring latent gene regulatory network kinetics. *Statistical applications in genetics and molecular biology*, 12(1):109–127, 2013.

Javier González, Ivan Vujačić, and Ernst Wit. Reproducing kernel Hilbert space based estimation of systems of ordinary differential equations. *Pattern Recognition Letters*, 45:26–32, 2014.

Peter J Green and Bernard W Silverman. *Nonparametric regression and generalized linear models: a roughness penalty approach*. CRC Press, 1993.

R Khanin, V Vinciotti, V Mersinias, CP Smith, and Ernst Wit. Statistical reconstruction of transcription factor activity using Michaelis–Menten kinetics. *Biometrics*, 63 (3):816–823, 2007.

Hua Liang and Hulin Wu. Parameter estimation for differential equation models using a framework of measurement error in regression models. *Journal of the American Statistical Association*, 103(484), 2008.

Alfred J Lotka. Analytical note on certain rhythmic relations in organic systems. *Proceedings of the National Academy of Sciences of the United States of America*, 6 (7):410, 1920.

Benn Macdonald, Catherine Higham, and Dirk Husmeier. Controversy in mechanistic modelling with Gaussian processes. In *Journal of Machine Learning Research: Workshop and Conference Proceedings*, volume 37, pages 1539–1547. Microtome Publishing, 2015.

Jim O Ramsay, G Hooker, D Campbell, and J Cao. Parameter estimation for differential equations: a generalized smoothing approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(5): 741–796, 2007.

Vladislav Vyshemirsky and Mark A Girolami. Bayesian ranking of biochemical system models. *Bioinformatics*, 24(6):833–839, 2008.

R. Wilkinson. Accelerating ABC methods using Gaussian processes. *Journal of Machine Learning Research - Workshop and Conference Proceedings: The 17th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 33:1015–1023, 2014.