Emulating complex dynamical simulators with random Fourier features

Hossein Mohammadi **Peter Challenor**

College of Engineering, Mathematics and Physical Sciences, University of Exeter, UK

Correspondence: h.mohammadi@exeter.ac.uk

Abstract: A methodology based on Gaussian process (GP) emulators is proposed that offers a fast approximation to complex dynamical models. Our method relies on emulating the short-time numerical flow map of the system, where the flow map is a function that returns the solution of a dynamical system at a certain time point, given initial conditions. Here, the flow map is emulated via a GP whose kernel is approximated with random Fourier features. This yields a random predictor whose realisations are approximations to the flow map. In order to predict a given time series (i.e., the model output), a single realisation of the approximate flow map is taken and used to iterate from the initial condition ahead in time. Repeating this procedure with multiple realisations from the distribution of approximate flow maps creates a distribution over the time series whose mean and variance serve as the model output prediction and the associated uncertainty, respectively. The proposed method is applied to emulate several dynamic non-linear computer codes. The results suggest that our approach has a high predictive performance and the associated uncertainty can capture the dynamics of the system accurately.

1 Gaussian process emulators

Let $f(\mathbf{x}), \mathbf{x} \in \mathcal{X}$, be the function we wish to emulate with the stochastic Gaussian process $Y(\mathbf{x})$ given by

$$Y(\mathbf{x}) = \mu(\mathbf{x}) + Z(\mathbf{x}). \tag{1}$$

Here, $\mu : \mathcal{X} \mapsto \mathbb{R}$ is the *trend function*, and $Z(\mathbf{x})$ is a centred (or zero mean) GP with the covariance function

$$\mathbb{C}\text{ov}\left(Z(\mathbf{x}), Z(\mathbf{x}')\right) = \sigma^2 k\left(\mathbf{x}, \mathbf{x}'\right), \ \forall \mathbf{x}, \mathbf{x}' \in \mathcal{X}. \tag{2}$$

The scalar σ^2 is the *process (signal) variance* and controls the scale of the amplitude of $Z(\mathbf{x})$. The correlation function $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is a *stationary (shift invariant)* kernel: $k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x} - \mathbf{x}')$. One of the most common stationary correlation functions is the squared exponential (SE) kernel defined as [4]

$$k_{SE}(\mathbf{x}, \mathbf{x}') = \exp\left(-0.5\left(\mathbf{x} - \mathbf{x}'\right)^\top \boldsymbol{\Delta}^{-2}\left(\mathbf{x} - \mathbf{x}'\right)\right), \tag{3}$$

where the diagonal matrix $\boldsymbol{\Delta} \in \mathbb{R}^{d \times d}$ consists of the *correlation length-scales* denoted by $\delta_1, \ldots, \delta_d$.

Let $\mathcal{D} = \{\mathbf{X}, \mathbf{y}\}$ denote the training data set in which $\mathbf{X} = [\mathbf{x}^1, \ldots, \mathbf{x}^n]^\top$ is called the *design matrix* and includes n points in the input space. The vector $\mathbf{y} = [f(\mathbf{x}^1), \ldots, f(\mathbf{x}^n)]^\top$ comprises the outputs at those locations. Typically, GPs are presented from a *function space* perspective where the predictive distribution relies on the posterior $Y(\mathbf{x}) \mid \mathcal{D}$. However, GPs can be interpreted from a *weight space* perspective in which $Z(\mathbf{x})$ is expressed by [4]

$$Z(\mathbf{x}) = \boldsymbol{\phi}(\mathbf{x})^\top \mathbf{w}, \quad \boldsymbol{\phi}(\mathbf{x}) = [\phi_1(\mathbf{x}), \phi_2(\mathbf{x}), \ldots]^\top, \tag{4}$$

that is a weighted sum of (possibly infinite) basis functions. The vector $\mathbf{w}$ consists of the weights and $\boldsymbol{\phi}(\mathbf{x}) : \mathcal{X} \mapsto \mathcal{H}$ is called the *feature map* and transforms the input space into a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ [3]. Given that all parameters in (1) are known, the predictive mean in the weight space view takes the following form

$$\tilde{f}(\mathbf{x}) = \mu(\mathbf{x}) + \boldsymbol{\phi}(\mathbf{x})^\top \boldsymbol{\Phi}\left(\boldsymbol{\Phi}^\top \boldsymbol{\Phi}\right)^{-1}(\mathbf{y} - \boldsymbol{\mu}), \tag{5}$$

where $\boldsymbol{\mu} = \mu(\mathbf{X})$ and $\boldsymbol{\Phi} = \left[\boldsymbol{\phi}(\mathbf{x}^1), \ldots, \boldsymbol{\phi}(\mathbf{x}^n)\right]$ is the aggregation of columns of $\boldsymbol{\phi}(\mathbf{x})$ for all points in the training set [4]. Equation (5) is obtained by assuming a Gaussian distribution over $\mathbf{w}$ and computing its posterior [4]. The weight space representation of GPs is particularly advantageous for purposes of drawing samples from their predictive mean. This is the driving idea behind the present work.

2 GP prediction with random Fourier features

Random Fourier features (RFF) offers an effective solution to approximate stationary kernels relying on Bochner's theorem [1]. According to this theorem, the Fourier transform of the stationary kernel k can be written as

$$k(\mathbf{x}, \mathbf{x}') = \int e^{-i\boldsymbol{\omega}^\top(\mathbf{x}-\mathbf{x}')} p(\boldsymbol{\omega}) d\boldsymbol{\omega} = \mathbb{E}_{p(\varphi)}\left[\varphi(\mathbf{x})^\top \varphi(\mathbf{x}')\right]. \tag{6}$$

A possible choice for $\varphi(\cdot)$ is

$$\varphi(\mathbf{x}) = \sqrt{2}\cos\left(\boldsymbol{\omega}^\top \mathbf{x} + b\right), \tag{7}$$

in which $b \sim \mathcal{U}[0, 2\pi]$ is a uniform random variable [3]. The distribution of $\boldsymbol{\omega}$ depends on the type of correlation function. For example, the spectral density of the SE kernel is Gaussian: $\boldsymbol{\omega}_{SE} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Delta}^{-2}\right)$.

The explicit random feature map $\varphi(\mathbf{x})$ defined by Equation (7) allows us to estimate the (actual) feature map $\boldsymbol{\phi}(\mathbf{x})$, which is possibly infinite dimensional. This can be performed using a Monte Carlo approach where we generate M independently and identically distributed (i.i.d.) samples from $p(\boldsymbol{\omega})$ and $p(b) = \mathcal{U}[0, 2\pi]$ denoted by $\boldsymbol{\omega}^{(1)}, \ldots, \boldsymbol{\omega}^{(M)}$ and $b^{(1)}, \ldots, b^{(M)}$, respectively. Then, the approximated feature map $\hat{\boldsymbol{\phi}}(\mathbf{x})$ is achieved by

$$\hat{\boldsymbol{\phi}}(\mathbf{x}) = \sqrt{\frac{2}{M}}\left[\cos\left(\boldsymbol{\omega}^{(1)\top}\mathbf{x} + b^{(1)}\right), \ldots, \cos\left(\boldsymbol{\omega}^{(M)\top}\mathbf{x} + b^{(M)}\right)\right]^\top, \tag{8}$$

which transforms an input vector $\mathbf{x}$ into the M-dimensional feature space. Finally, the stationary kernel k is approximated as

$$k(\mathbf{x}, \mathbf{x}') = \mathbb{E}_{p(\varphi)}\left[\varphi(\mathbf{x})^\top \varphi(\mathbf{x}')\right] \approx \hat{\boldsymbol{\phi}}(\mathbf{x})^\top \hat{\boldsymbol{\phi}}(\mathbf{x}'). \tag{9}$$

The predictive mean of a GP whose kernel is approximated with RFF takes the following form

$$\hat{f}(\mathbf{x}) = \mu(\mathbf{x}) + \hat{\boldsymbol{\phi}}(\mathbf{x})^\top \hat{\boldsymbol{\Phi}}\left(\hat{\boldsymbol{\Phi}}^\top \hat{\boldsymbol{\Phi}}\right)^{-1}(\mathbf{y} - \boldsymbol{\mu}), \tag{10}$$

wherein $\hat{\boldsymbol{\Phi}} = \left[\hat{\boldsymbol{\phi}}(\mathbf{x}^1), \ldots, \hat{\boldsymbol{\phi}}(\mathbf{x}^n)\right]$ is an $M \times n$-dimensional matrix and estimates $\boldsymbol{\Phi}$. Since the construction of $\hat{\boldsymbol{\phi}}(\mathbf{x})$ (see Equation (8)) relies on i.i.d. samples taken from the kernel spectral density $p(\boldsymbol{\omega})$ and $p(b)$, the predictive mean in (10) is stochastic and its s-th realisation is denoted by $\hat{f}^{(s)}(\mathbf{x})$. By repeating the Monte Carlo sampling scheme, we can generate multiple such realisations. This idea is used to emulate dynamical simulators where multiple draws from the approximate flow map are employed to perform one-step ahead predictions. With this, one can quantify uncertainty of the time series prediction in the absence of a closed-form expression.

3 Emulating dynamical simulators

Let $\mathbf{x}(t_1) = (x_1(t_1), \ldots, x_d(t_1))^\top$ denote the solution of the system at $t_1 = t_0 + \Delta t$ for a given fixed "small" time step Δt and initial condition $\mathbf{x}_0$. We assume that $\mathbf{x}(t_1)$ is produced by the flow map F defined as

$$F(\mathbf{x}_0) = (f_1(\mathbf{x}_0), \ldots, f_d(\mathbf{x}_0))^\top = (x_1(t_1), \ldots, x_d(t_1))^\top, \tag{11}$$

such that each map $f_i : \mathcal{X} \mapsto \mathbb{R}$ yields the i-th component of $\mathbf{x}(t_1)$, i.e. $x_i(t_1)$. A prediction associated with the dynamics of $x_i(t)$ is achieved by:

- Emulating f_i by a GP with RFF approximation, $\hat{f}_i$
- Drawing a realisation from $\hat{f}_i$ denoted by $\hat{f}^{(s)}$
- Using $\hat{f}^{(s)}$ iteratively to perform one-step ahead predictions

Following the above procedure renders only one prediction to the time series. However, we wish to have an estimation of uncertainty associated with the prediction accuracy. This can be achieved by repeating the above steps with different draws from the emulated flow map to reach a distribution over the time series. The mean and variance of that distribution then serve as the model output prediction and the associated uncertainty, respectively.

4 Numerical results

The prediction performance of our proposed method is tested on several dynamical systems implemented as computer codes. They are the Lorenz, van der Pol [5], and Hindmarsh-Rose [2] models. The latter simulates the dynamics of a single neuron. The results are shown in the figures below.

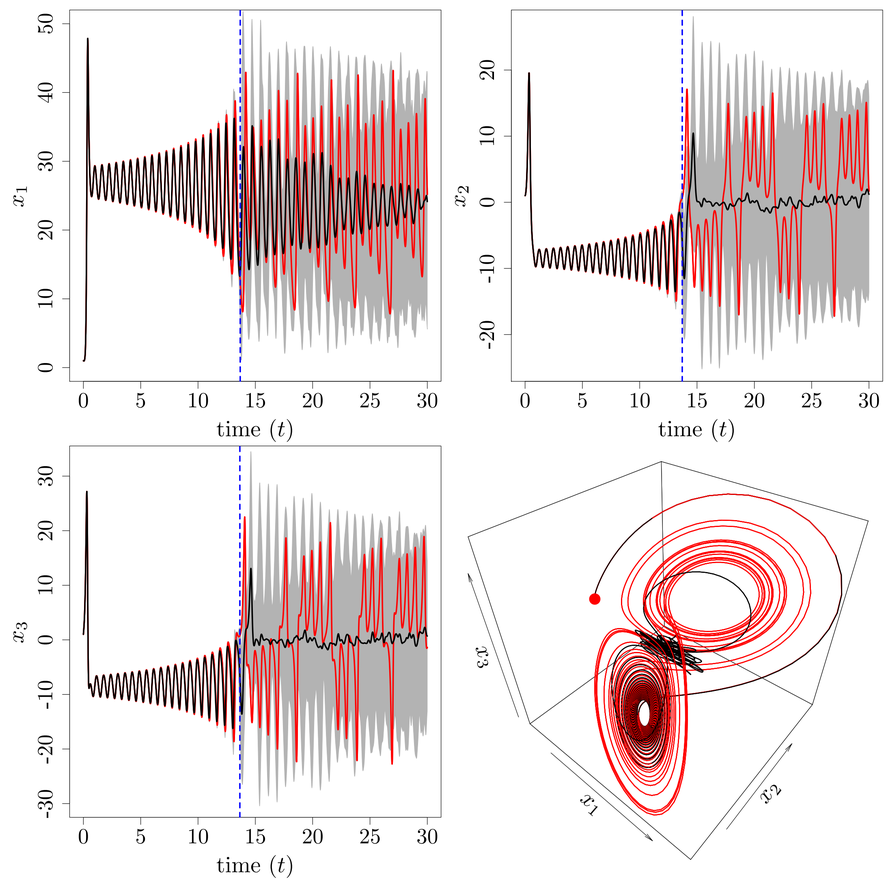

Figure 1: The prediction (black) and associated uncertainty (shaded) in emulating the Lorenz model (red). The initial condition is $\mathbf{x}_0 = (1, 1, 1)^\top$ denoted by a red point in the three-dimensional picture. The vertical dashed blue lines represent the predictability horizon which are the change point in the diagram of prediction uncertainties.

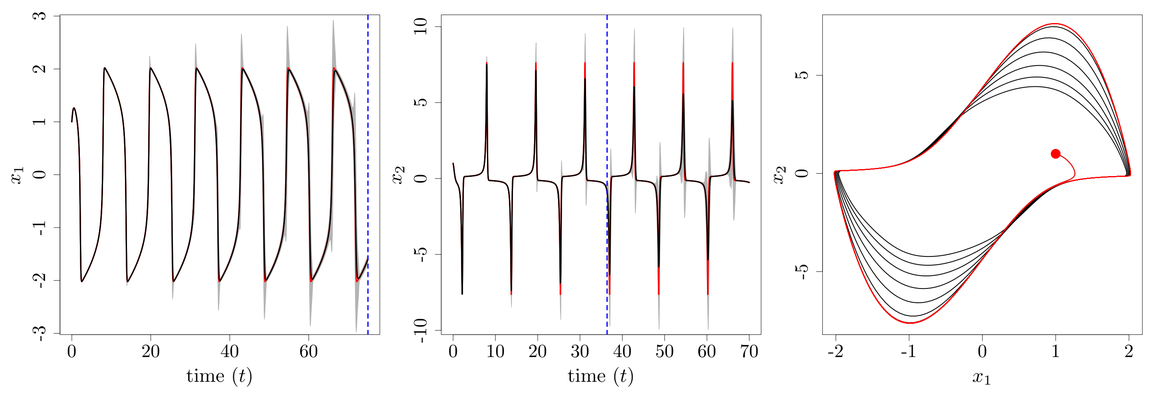

Figure 2: The van der Pol oscillator (red), the prediction (black) and credible interval (shaded). The dashed blue lines show the predictability horizon. The initial condition (red point) is $\mathbf{x}_0 = (1, 1)^\top$.

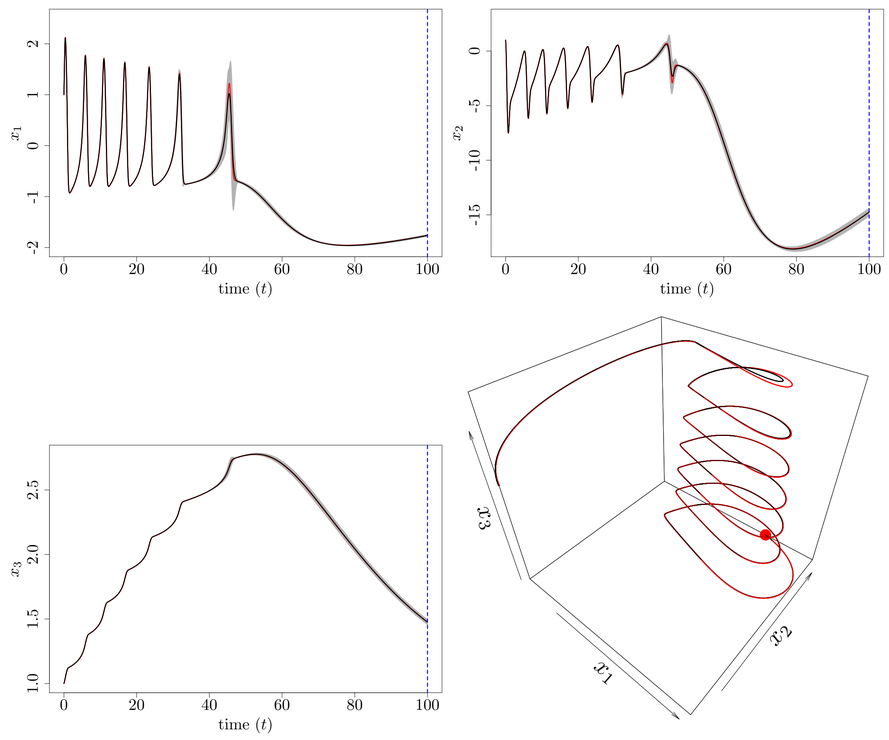

Figure 3: The prediction (black) and associated uncertainty (shaded) in emulating the Hindmarsh-Rose model (red). The initial condition is $\mathbf{x}_0 = (1, 1, 1)^\top$. The vertical dashed blue lines represent the predictability horizon which are the change point in the diagram of prediction uncertainties.

References

[1] Salomon Bochner. *Lectures on Fourier integrals*. Princeton University Press, 1959.

[2] J. L. Hindmarsh and R. M. Rose. A model of neuronal bursting using three coupled first order differential equations. *Proceedings of the Royal Society of London*, 221(1222):87–102, 1984.

[3] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, pages 1177–1184. Curran Associates Inc., 2008.

[4] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian processes for machine learning (adaptive computation and machine learning)*. The MIT Press, 2005.

[5] S.H. Strogatz. *Nonlinear dynamics and chaos*. Studies in nonlinearity. Sarat Book House, 2007.