Unsupervised Learning with Gaussian Processes and Stochastic Differential Equations

David Fernandes [1]   Carl Henrik Ek [2]   Neill D. F. Campbell [1]

[1]University of Bath, UK   [2]University of Cambridge, UK

Introduction

- With the massive increase in the amount of data, not all of it can be labelled
  - Unsupervised methods are used to group data or find lower-dimensional representations of them (or both)
- Deep Learning has played a major role in this, but
  - Building deep learning models is more art than science
  - No uncertainty propagation - learned mappings are deterministic
- These problems have been tackled in many ways
  - Differential equations (e.g., NeuralODE) partly free us from architecture design
  - Gaussian processes (GPs), Bayesian neural networks and, more recently, Stochastic Differential Equations (SDEs) can be used to introduce stochasticity in the model
- Our model seeks to do unsupervised learning while avoiding these problems
  - Specifically, it seeks to find a lower-dimensional representation of time-series data
  - Our model uses both GPs and SDEs

Desired properties

- The model should be able to find a lower-dimensional representation of time-series data
- It should be possible to generate new data points from this lower-dimensional manifold
- Epistemic uncertainty is propagated throughout the whole model
  - The model should be more uncertain of its outputs the further away from the training data points we are

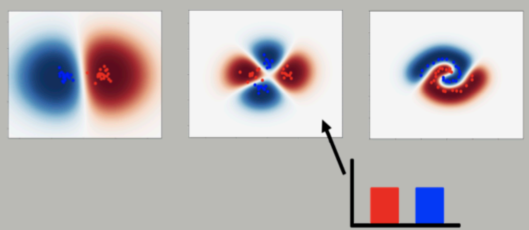

Figure: Example of model with epistemic uncertainty.

- The model must be able to handle irregularly-sampled time-series
  - Many realistic data sets are not obtained at regular time intervals

Model Overview

- Uses General Itô's formula (see below)
  - Relates two stochastic processes with different dimensionalities, through a deterministic mapping that connects those two spaces
  - Ideal for obtaining lower-dimensional dynamics ($\boldsymbol{X}$ in the formula) of the data ($\boldsymbol{Y}$)
- Lower-dimensional SDE is modelled using a GP
  - Both the drift function $\boldsymbol{u}$ and $\boldsymbol{v}$ are obtained from a GP
  - Possible to impose certain conditions on these functions with the GP kernel
  - Since we can choose the dimensionality of the latent space, we can avoid the difficulties of constructing a GP in a high-dimensional space
  - Having this lower-dimensional space, it is then possible to generate new data points from it
- Freedom to choose the deterministic mapping that connects the two spaces (as long as it is twice differentiable)
  - From simple linear combinations to neural networks
  - If there is a suspected relationship between the two spaces, this can be encoded here
- Uncertainty is automatically accounted for in both spaces, as the data is modelled by SDEs in both of them
  - This applies irrespectively of the mapping chosen, so even when using deterministic mappings (such as neural networks), there is uncertainty in the model
- Can deal with irregularly-timed sampled data
  - Time-steps used to solve SDEs can be adjusted

Theorem (General Itô's formula)

Let

$$d\boldsymbol{X}(t) = \boldsymbol{u}dt + \boldsymbol{v}d\boldsymbol{B}(t)$$

be an $\boldsymbol{n}$-dimensional Itô process. Let $\boldsymbol{g}(t, \boldsymbol{x}) = (g_1(t, \boldsymbol{x}), \dots, g_p(t, \boldsymbol{x}))$ be a C^2 map from $[0, \infty) \times \mathbb{R}^n$ into $\mathbb{R}^p$.
Then the process

$$\boldsymbol{Y}(t, w) = \boldsymbol{g}(t, \boldsymbol{X}(t))$$

is again an Itô process, whose component number $\boldsymbol{k}$, $\boldsymbol{Y_k}$, is given by

$$dY_k = \frac{\partial g_k}{\partial t}(t, X)\, dt + \sum_i \frac{\partial g_k}{\partial x_i}(t, X)\, dX_i + \frac{1}{2}\sum_{i,j} \frac{\partial^2 g_k}{\partial x_i \partial x_j}(t, X)\, dX_i\, dX_j$$

where $dB_i\, dB_j = \delta_{ij}\, dt$, $dB_i\, dt = dt\, dB_i = 0$.

Preliminary results

- Tested with an illustrative data set
  - Latent space is a dual moon that rotates and gets noisier
  - Observed space is the latent space multiplied by a $\boldsymbol{8 \times 2}$ random matrix
  - Only the initial and final time-steps of the observed space are provided

$t_{\text{step}} = 0$ $t_{\text{step}} = 15$ $t_{\text{step}} = 30$

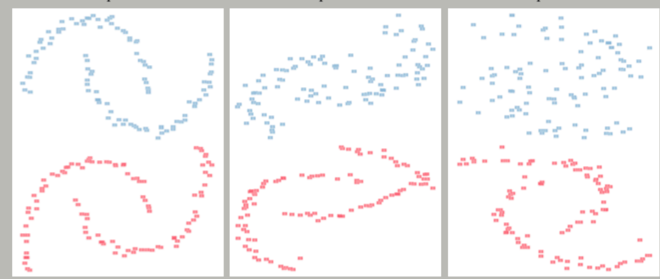

Figure: Top (blue): The (unobserved) 2D input space used to generate the noisy 8D data. Bottom (red): The latent space inferred by the model for three different time steps of the stochastic process. As time increases, the input space rotates anti-clockwise (which the model is able to learn), while the noise increases (which the model is able to remove).

Next steps

- Test if model is able to handle different types of data
  - Varying frequency (chirp)
  - Changes in the number of modes

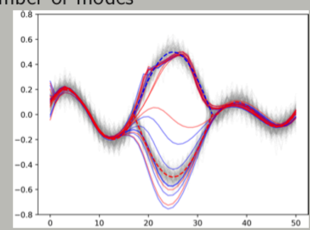

Figure: Example of a function with changes in the number of modes.

- Decrease training time

References

[1] B. Øksendal. *Stochastic Differential Equations: An Introduction with Applications*. Hochschultext / Universitext. Springer, 2003.

[2] Constantino Antonio García Martínez. *A Bayesian approach to simultaneously characterize the stochastic and deterministic components of a system*. PhD thesis, 2019.

[3] Simo Särkkä and Arno Solin. *Applied Stochastic Differential Equations*. Institute of Mathematical Statistics Textbooks. Cambridge University Press, 2019.

[4] Tian Qi Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 6571–6583. Curran Associates, Inc., 2018.

[5] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2005.

dlf28@bath.ac.uk