One shot learning of stochastic differential equations with Gaussian Processes and computational graph completion.

M. Darcy[1] B. Hamzi[1,2] G. Livieri[3] H. Owhadi[1] P. Tavallali[4]

[1]California Institute of Technology

[2]Johns Hopkins University [3]Scuola Normale Superiore [4]JPL, NASA

ICCOPT July 2022

Table of Contents

Problem statement

We consider stochastic differential equation (SDE) of the form:

$$dX_t = f(X_t)dt + \sigma(X_t)dW_t, \quad X_0 = x_0$$

where W_t is a Brownian motion and

$$f : \mathbb{R} \to \mathbb{R} \quad \text{drift}$$
$$\sigma : \mathbb{R} \to \mathbb{R} \quad \text{diffusion}$$

are unknown functions.

Objective

Recover the drift f and diffusion σ given a finite number of observations coming from a *single sample trajectory* $X := (X_{t_n})_{n=1}^N$ separated by time-steps Δt_n,

Motivation and challenges

Motivation

SDEs allow us to model systems subject to random effects and have applications in finance, dynamical systems, engineering ...

The problem we consider is challenging:

- The observations X come from a single trajectory.
- We make few assumptions on f and σ.
- The observations X only provide indirect information on f and σ.
- The sampling time-steps Δt_n introduce a discretization error.

Method summary

Our method can be summarized as follows

1. Formulate our model as a **computational graph** with unknown functions.
2. Recover the functions using Gaussian processes by **completing the graph**.
3. **Optimize the hyper-parameters** of the Gaussian processes using cross-validation.

Our method allows us to

- **Recover** f, σ at observed points (hard).
- **Forecast** future values of f, σ (harder).

Modeling Assumption

Let $X_n := X_{t_n}$. We assume the following discretization, the Euler-Maruyama model given

$$X_{n+1} = X_n + f(X_n)\Delta t_n + \sigma(X_n)\sqrt{\Delta t_n}\xi_n + \varepsilon_n$$

where

$$\xi_n \overset{d}{\sim} \mathcal{N}(0,1) \quad \text{dynamics noise}$$
$$\varepsilon_n \overset{d}{\sim} \mathcal{N}(0,\lambda) \quad \text{modeling noise}$$

are independent.
Defining $Y_n := X_{n+1} - X_n$, our model can be restated as

$$\boxed{Y_n = f(X_n)\Delta t_n + \sigma(X_n)\sqrt{\Delta t_n}\xi_n + \varepsilon_n}$$

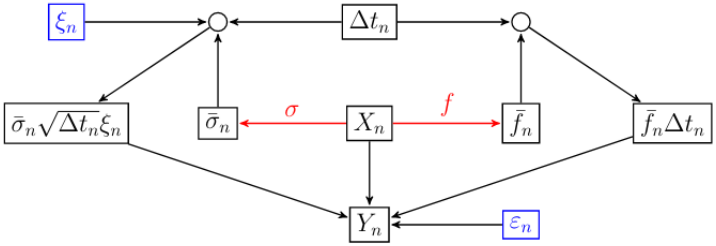
Table of Contents

Computational Graph

The model

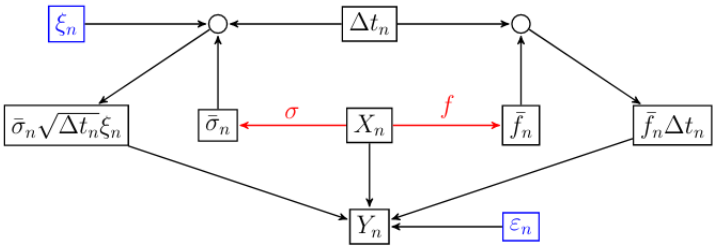
$$Y_n = f(X_n)\Delta t_n + \sigma(X_n)\sqrt{\Delta t_n}\xi_n + \varepsilon_n$$

can be re-stated as a computational graph:



Computational Graph

$$Y_n = f(X_n)\Delta t_n + \sigma(X_n)\sqrt{\Delta t_n}\xi_n + \varepsilon_n$$



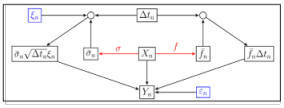
- Arrows $\rightarrow$ represent functions.
- Nodes $\circ$ aggregate incoming variables.
- X_n and Y_n are input and outputs.
- Blue variables are noise.
- Red variables are unknown functions we wish to recover.

The Computational Graph Approach

Model

$$Y_n = f(X_n)\Delta t_n + \sigma(X_n)\sqrt{\Delta t_n}\xi_n + \varepsilon_n$$

Graph



Computational Graph Completion (CGC) [Owh21] proposes to replace the unknown functions f and σ by Gaussian processes and to recover them by Maximum A Posteriori (MAP) estimation given inputs and outputs of the graph $(X_n, Y_n)_{n=1}^N$.

We "forget" about the underlying model, we consider the data $(X_n, Y_n)_{n=1}^N$ to be inputs and outputs of the graph.

Gaussian process prior

We assume that f and σ are distributed according to **independent** Gaussian processes:

$$f \overset{d}{\sim} \mathcal{GP}(\mathbf{0}, \boldsymbol{K})$$
$$\sigma \overset{d}{\sim} \mathcal{GP}(\mathbf{0}, \boldsymbol{G}).$$

Definition

A function f is distributed according to a Gaussian Process with covariance function (kernel) $\boldsymbol{K} : \mathbf{R} \times \mathbf{R} \to \mathbf{R}$ if

$$f(X) = (f(X_1), f(X_2), \ldots, f(X_n)) \overset{d}{\sim} \mathcal{N}(0, K(X, X))$$

where $K(X, X) \in \mathbf{R}^{n \times n}$ with entries $K(X, X)_{ij} = \boldsymbol{K}(X_i, X_j)$.

The kernel function $\boldsymbol{K}$ is often parameterized by some parameter $\boldsymbol{\theta}$.

Recovery of the drift and diffusion

The recovery of f and σ can be separated into two steps

1. **Recover** the values of f and σ at observed data points $(X_n)_{n=1}^N$.
2. **Forecast** future values of f and σ using the recovered values.

We use $\bar{f} \in \mathbb{R}^N$ and $\bar{\sigma} \in \mathbb{R}^N$ to denote the function values at the observed data points:

$$\bar{f}_n := f(X_n)$$
$$\bar{\sigma}_n := \sigma(X_n).$$

MAP estimation

We must first recover $\bar{f} \in \mathbb{R}^N$ and $\bar{\sigma} \in \mathbb{R}^N$. By Bayes' rule

$$p(\bar{f}, \bar{\sigma} | Y, X) = p(Y | \bar{f}, \bar{\sigma}) \frac{\overbrace{p(\bar{f}|X)p(\bar{\sigma}|X)}^{\text{independence}}}{p(Y|X)}.$$

As is standard, we consider the negative log likelihood. The recovery of $\bar{f}$ and $\bar{\sigma}$ is given as the solution to the problem

$$\bar{f}^*, \bar{\sigma}^* = \arg\min_{\bar{f}, \bar{\sigma}} - \ln(p(\bar{f}, \bar{\sigma} | Y, X)). \tag{1}$$

MAP estimation

Using our graph and our prior on $\bar{f}$ and $\bar{\sigma}$:

$$-\ln p(\bar{f},\bar{\sigma}|Y,X) \propto \mathcal{L}(\bar{f},\bar{\sigma}) := \overbrace{(Y-\Lambda\bar{f})^T(\Sigma+\lambda I)^{-1}(Y-\Lambda\bar{f}) + \sum_{n=1}^{N}\ln(\bar{\sigma}_n^2\Delta t_n+\lambda)}^{-\ln p(Y|\bar{f},\bar{\sigma})} + \underbrace{\bar{f}^T K(X,X)^{-1}\bar{f}}_{-\ln p(\bar{f}|X)} + \underbrace{\bar{\sigma}^T G(X,X)^{-1}\bar{\sigma}}_{-\ln p(\bar{\sigma}|X)}.$$

where Σ is a diagonal matrix with entries $\bar{\sigma}_n^2\Delta t_n$, and Λ is a diagonal matrix with entries Δt_n.

The recovery of f, σ is reduced to the minimization of $\mathcal{L}(\bar{f},\bar{\sigma})$.

Alternative minimization

Representer theorem

For any given $\bar{\sigma}$, the minimizer in $\bar{f}$ of $\mathcal{L}(\bar{f}, \bar{\sigma})$ is

$$\bar{f}^*(\sigma) := \arg\min_{\bar{f}} \mathcal{L}(\bar{f}, \bar{\sigma}) = K(X,X)\Lambda\Big(\Lambda K(X,X)\Lambda + \Sigma + \lambda I\Big)^{-1} Y$$

Using the representer theorem, and plugging $f^*(\sigma)$ into the original loss, the minimization in σ is:

$$\mathcal{L}(\bar{f}^*(\sigma), \bar{\sigma}).$$

The function $\mathcal{L}(\bar{f}^*(\sigma), \bar{\sigma})$ is non-convex and difficult to minimize. We use a gradient descent based method with an adaptive step size and momentum.

Table of Contents

Motivation

Learning the hyper-parameters $\boldsymbol{\theta}$ of the kernel functions $\boldsymbol{K}$ and $\boldsymbol{G}$ can drastically improve the performance of the recovery and prediction.

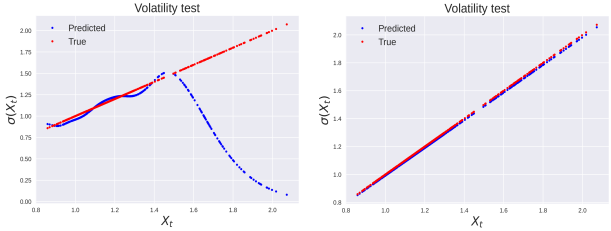


Figure: Two forecasts: non-learned kernel (left) and data learned kernel (right).

To this end, we use a randomized cross-validation approach to learn the kernels from data.

General principle

Randomized cross validation for kernel hyper-parameters relies on two principles

- **Cross validation:** optimize the model on a subset $\mathcal{D}_\Pi$ of the data and measure the performance on a withheld subset $\mathcal{D}_{\Pi^c}$, using some metric $\mathcal{L}_{\text{CV}}$.
- **Randomized:** as proposed in [OY19], sample subsets $(\mathcal{D}_\Pi, \mathcal{D}_{\Pi^c})$ randomly and use this noisy loss to optimize the hyperparameters $\boldsymbol{\theta}$.

The use of a random samples often leads to a choice of hyper-parameters $\boldsymbol{\theta}$ which is more robust.
We iteratively sample cross-validation sets and select the best parameters $\boldsymbol{\theta}$ using a Bayesian optimization algorithm to minimize the

$$\mathcal{L}_{\text{CV}}(\boldsymbol{\theta}; \bar{f}_i^*, \bar{\sigma}_i^*, \mathcal{D}_{\Pi_i}) = -\ln p(Y_{\Pi_i} | \bar{f}^*, \bar{\sigma}^*).$$

Table of Contents

Numerical results

For our numerical experiments, we consider several systems

$dX_t = \sin(2k\pi X_t)dt + b\cos(2k\pi X_t)dW_t$	Trigonometric process.
$dX_t = \mu X_t dt + b\exp(-X_t^2)dW_t$	Exponential decay volatility.
$dX_t = \mu X_t dt + \sigma X_t dW_t$	Geometric Brownian motion (GBM).

We use the Matérn Kernel with smoothness parameter $\nu = \frac{5}{2}$:

$$K_{\text{Matern}}(x,y) = \sigma^2\Big(1 + \frac{\sqrt{5}||x-y||}{l} + \frac{5||x-y||^2}{3l^2}\Big)\exp\Big(-\frac{\sqrt{5}||x-y||}{l}\Big).$$

In this case, the parameters are $\boldsymbol{\theta} = (\sigma, l)$.

Trigonometric process

$$dX_t = \sin(2k\pi X_t)dt + b\cos(2k\pi X_t)dW_t \quad \text{Trigonometric process.}$$

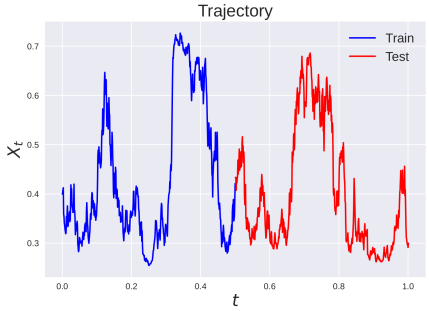


Figure: The trigonometric process.

Trigonometric process

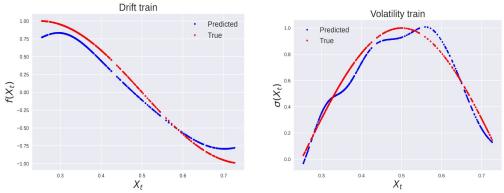


Figure: Recovery of the drift and diffusion

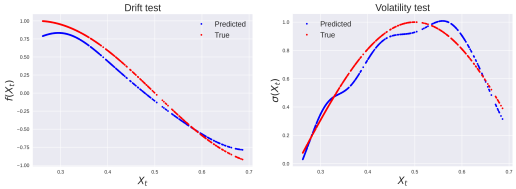


Figure: Forecast of the drift and diffusion.

Exponential decay volatility

$$dX_t = \mu X_t dt + b \exp(-X_t^2) dW_t \quad \text{Exponential decay volatility.}$$

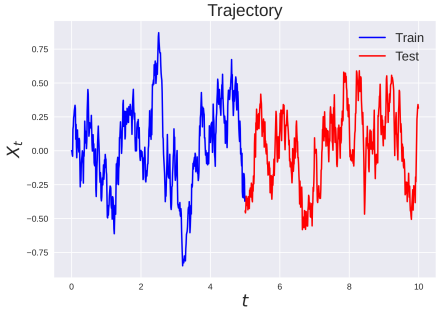


Figure: Exponential decay volatility process

Exponential decay volatility

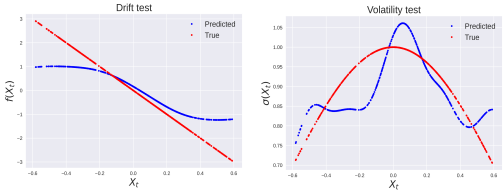


Figure: Forecast: non-learned kernels.

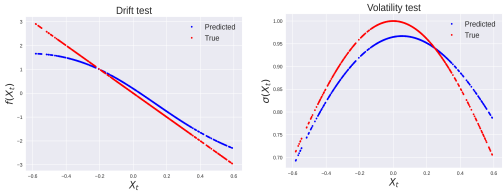


Figure: Forecast: learned kernel.

GBM

$$dX_t = \mu X_t dt + \sigma X_t dW_t \quad \text{Geometric Brownian motion (GBM).}$$

Figure: Geometric Brownian motion

GBM

$$dX_t = \mu X_t dt + \sigma X_t dW_t \quad \text{Geometric Brownian motion (GBM).}$$

For GBM, the linear kernel

$$K_{\text{linear}}(x, y) = \sigma^2(x^\intercal y + c)$$

is better specified than the Matérn kernel. Optimizing the hyper-parameters yields similar performance to a well specified kernel.

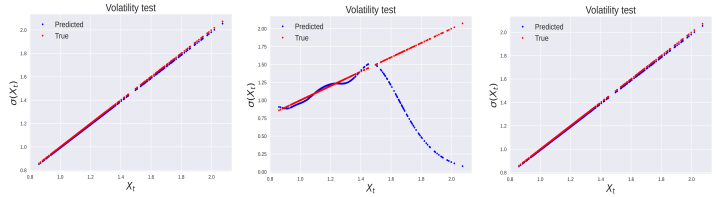


Figure: Forecast: linear kernel, non-learned kernel and learned kernel.

Contributions

The proposed methods

- Provide a general framework to recover the drift and diffusion of SDEs using a small number of observations.
- Provide a framework to optimize the parameters of covariance functions for this problem.

Preprint available (to be updated) [Dar+22]

mdarcy@caltech.edu

Bibliography

[Dar+22] Matthieu Darcy et al. *One-Shot Learning of Stochastic Differential Equations with Computational Graph Completion*. Feb. 2022. DOI: 10.13140/RG.2.2.32262.65604. URL: https://www.researchgate.net/publication/358263232_One-Shot_Learning_of_Stochastic_Differential_Equations_with_Computational_Graph_Completion.

[Owh21] Houman Owhadi. *Computational Graph Completion*. 2021. DOI: 10.48550/ARXIV.2110.10323. URL: https://arxiv.org/abs/2110.10323.

[OY19] Houman Owhadi and Gene Ryan Yoo. "Kernel Flows: From learning kernels from data into the abyss". In: *Journal of Computational Physics* 389 (2019), pp. 22–47. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2019.03.040. URL: https://www.sciencedirect.com/science/article/pii/S0021999119302232.