M2 Internship: Physics Informed Neural Networks for parameter estimation in Stochastic Differential Equations

Contacts:
Lucia Clarotto - lucia.clarotto@agroparistech.fr
Sophie Donnet - sophie.donnet@inrae.fr
Hugo Gangloff - hugo.gangloff@inrae.fr
Nicolas Jouvin - nicolas.jouvin@inrae.fr

Keywords: PINNs, SDEs, Generative models, spatial ecology

Deadline for application: February 1st, 2026

To apply send an email with object "Application for SDE-PINN internship" with your CV, an academic transcript for M1, and a motivation letter.

1 Context

Stochastic Differential Equations (SDEs) are popular models in many fields including spatial ecology (Michelot et al., 2019), climate science (Ditlevsen and Ditlevsen, 2023), biology (Degond et al., 2020). Diffusion SDEs with additive noise are commonly found and defined as in Øksendal (2003):

$$\mathrm{d}X_t = F(X_t;\beta)\,\mathrm{d}t + \Sigma\,\mathrm{d}W_t, \quad X_0 \sim p_0, \tag{1}$$

where $(X_t)_t \in \mathbb{R}^d$ is the stochastic process of interest, W_t is a d-dimensional Brownian noise, p_0 is the initial distribution, β parametrizes the drift of the equation and Σ is the diffusion coefficient.

When proposing such a model for observed trajectories at discrete times $(x_{t_1}, \dots, x_{t_n})$, the next step consists in estimating the parameter $\theta = \{\beta, \Sigma\}$ from those observed data. This is a critical task from which one can gain understanding on the underlying process mechanics.

One classical parameter estimation approach is that of maximum likelihood. In some rare cases, when the SDE is such that the likelihood of the observations can be computed explicitly as a function of the parameters θ, the estimation then resorts to a classical estimation task. However, in many cases this approach is not possible, and a classical procedure is to locally linearize the EDS. Many approaches have been proposed over the last decades, all with strengths and weaknesses (Pilipovic et al., 2024).

To the extent of our knowledge, another feature of SDEs has been under-used in the estimation context. Indeed, let $p(x,t;\theta)$ be the density function of X_t for a given set of parameters θ. The behavior of $p(x,t;\theta)$ is described by the Fokker-Planck Equation (FPE) (Risken, 1989), which is the Partial Differential Equation (PDE) defined by

$$\begin{aligned}\frac{\partial p(x,t;\theta)}{\partial t} &= -\nabla \cdot \left(F(x;\beta)p(x,t;\theta)\right) + \frac{1}{2}\nabla \cdot \left(\Sigma\Sigma^\top \nabla p(x,t;\theta)\right),\\ p(\cdot,0;\theta) &= p_0\end{aligned}$$

where ∇ and $\nabla\cdot$ denotes the gradient and divergence operators. Thus, solving this PDE would provide an implicit expression of the marginal likelihood of each observation x_{t_i}, which is a first step towards the maximum likelihood estimation of θ.

In the past few years, the emergence of Physics-Informed Neural Networks (PINNs) (Raissi et al., 2019) has led to a fundamental rethinking of traditional approaches to solving partial differential equations. In a few words, the PINNs approach seeks to find the best neural network u_ν (ν being the set of weights and biases) representing the solution of the PDE in the form $\mathcal{N}_\theta[u] = 0$, where $\mathcal{N}_\theta$ is an arbitrary differential operator, by minimizing its residuals computed at randomly sampled collocation points, in a so-called forward problem. This mesh-less approach has proven useful in a variety of contexts. It can also be extended to inverse problems where one seeks to learn the differential operator's parameters θ given some observations of the solution $p(x_i, t_j; \theta)$, thus offering a flexible way to incorporate available "data" in the training.

Two additional difficulties arise in the context of this internship:

1. First, $p(x, t; \theta)$ being a density function, the PINN is expected to learn a normalized probability density, hence one must ensure that, for any t, $\int_\Omega p(x, t; \theta)\mathrm{d}x = 1$.
2. Second, we do not observe the solution itself but realisations of the SDE at discrete time points, whose marginal distribution is the solution of the PDE.

A recent line of research uses PINNs for simulation or parameter estimation in SDEs via their FPE (Feng et al., 2021; Chen et al., 2020; Liu et al., 2023), as we have just described. In this context, building on the previous articles, *this internship will explore the connection between SDEs, their FPE, and the Physics-Informed Neural Network (PINN) methodology.*

2 Goal of this internship: parameter estimation in SDEs with PINNs

The internship aims at proposing an efficient neural network architecture and optimisation scheme to accurately solve a FPE (forward problem) and perform parameter estimation (inverse problem) by using observational data that are assumed to be generated by the corresponding SDE. Since the solution to the FPE is a normalized probability density, an interesting line of research considers using Normalizing Flows (NFs) (Papamakarios et al., 2021), as these architectures inherently encode the normalization constraint of probability densities. The Temporal NF (Both and Kusters, 2019) with KR-net (Feng et al., 2021; Tang et al., 2022) seems particularly well suited for this task. Such an architecture has been combined with a new loss function for training the PINN in Bekri et al. (2025), where the author proposes to switch from the standard loss function of PINNs to a loss function involving the Feynman-Kac formula:

$$p(x, t) = \mathbb{E}\left[\exp\left(-\int_0^t q(\tilde{X}_s, s)\mathrm{d}s\right) f(\tilde{X}_t)|\tilde{X}_0 = x\right], \tag{2}$$

where the definitions of the function $q(x, s)$ and the stochastic process $\{\tilde{X}_t\}$ can be deduced from Equation 2.

The advantage resides in the fact that it has been shown to be well suited for non-stationary FPEs and to resolve some of the convergence issues of the vanilla PINN framework (Mandal and Apte, 2024; Bekri et al., 2025). In this internship, we plan to test those new sophisticated approaches, since the classical PINN framework fails on more intricate FPEs.

Concerning the modeling approach, we could also consider using the FPE for the logarithm of the probability distribution (Hu et al., 2025). Such an approach is another way, along NFs, to alleviate the burden of the normalizing constant. It also draws fruitful links with diffusion models (Lai et al., 2023).

An important part of the internship resides in the comparison of the developed PINN approach with other state-of-the-art approaches for parameter estimation in SDEs (Michelot et al., 2019; Pilipovic et al., 2024). Despite the fact that a proposed PINN model would lack theoretical guarantees (such as convergence guarantees), we expect that the PINN exhibits better accuracy in the estimation for a reduced computational time (Hu et al., 2025). This should be particularly true for high dimensional stochastic processes, as PINN training via the FPE does not require to linearize the equation, and benefits from optimized computations on GPUs. The validation of the models and parameter estimation approaches will first be carried out on synthetic data before considering observational data from spatial ecology or climate sciences.

Organization The internship will be organized around the following tasks:

- Bibliography on the recent literature on PINNs for parameter estimation in SDEs;
- Conception of a new PINN model for parameter inference in SDEs;
- Implementation of the new model using the `jinns` (Gangloff and Jouvin, 2024) Python library, developed at MIA Paris-Saclay and based on the JAX ecosystem[1];
- Comparison between the new model and classical approaches on synthetic and real-world data.

3 Profile & environment

The candidate should be a 2nd year master or last year engineer student, in statistics, machine learning, stochastic processes, computational statistics or deep learning. Scientific programming skills in Python are required, while familiarity with the JAX ecosystem and R language is a bonus.

- Location: UMR MIA Paris-Saclay, Palaiseau Campus, 22 place de l'agronomie, 91120 Palaiseau, France
- Supervision: Lucia Clarotto is a researcher in spatial statistics. Sophie Donnet is a specialist in statistical modeling, with a specific focus on applications in ecology. Hugo Gangloff and Nicolas Jouvin do their research on PINNs and are the developers of the `jinns` Python package.
- Starting date: flexible, ideally starting in early 2026.
- Duration: 6 months
- Salary: as an intern, you'll receive a "gratification" which is unfortunately capped around 650 euros/month.

The candidate will have an office, and benefit from the work environment of the MIA Paris-Saclay laboratory, with many PhD students & postdocs working on statistical modeling and machine learning for the life sciences.

References

Bekri, N. E., Drumetz, L., and Vermet, F. (2025). Flowkac: An efficient neural fokker-planck solver using temporal normalizing flows and the feynman kac-formula. *arXiv preprint arXiv:2503.11427*.

Both, G.-J. and Kusters, R. (2019). Temporal normalizing flows. *arXiv preprint arXiv:1912.09092*.

Chen, X., Yang, L., Duan, J., and Karniadakis, G. E. (2020). Solving inverse stochastic problems from discrete particle observations using the fokker-planck equation and physics-informed neural networks.

Degond, P., Herda, M., and Mirrahimi, S. (2020). A Fokker-Planck approach to the study of robustness in gene expression. *Mathematical Biosciences and Engineering*, 17(6):6459–6486.

[1] https://jax.readthedocs.io/en/latest/

Ditlevsen, P. and Ditlevsen, S. (2023). Warning of a forthcoming collapse of the atlantic meridional overturning circulation. *Nature Communications*, 14(1):1–12.

Feng, X., Zeng, L., and Zhou, T. (2021). Solving time dependent fokker-planck equations via temporal normalizing flow. *arXiv preprint arXiv:2112.14012*.

Gangloff, H. and Jouvin, N. (2024). jinns: a jax library for physics-informed neural networks. *arXiv preprint arXiv:2412.14132*.

Hu, Z., Zhang, Z., Karniadakis, G. E., and Kawaguchi, K. (2025). Score-based physics-informed neural networks for high-dimensional fokker–planck equations. *SIAM Journal on Scientific Computing*, 47(3):C680–C705.

Lai, C.-H., Takida, Y., Murata, N., Uesaka, T., Mitsufuji, Y., and Ermon, S. (2023). Fp-diffusion: Improving score-based diffusion models by enforcing the underlying score fokker-planck equation. In *International Conference on Machine Learning*, pages 18365–18398. PMLR.

Liu, F., Wu, F., and Zhang, X. (2023). Pinf: Continuous normalizing flows for physics-constrained deep learning.

Mandal, P. and Apte, A. (2024). Solving fokker-planck equations using the zeros of fokker-planck operators and the feynman-kac formula. *arXiv preprint arXiv:2401.01292*.

Michelot, T., Gloaguen, P., Blackwell, P. G., and Etienne, M.-P. (2019). The langevin diffusion as a continuous-time model of animal movement and habitat selection. *Methods in Ecology and Evolution*, 10(11).

Øksendal, B. (2003). Stochastic differential equations. In *Stochastic differential equations: an introduction with applications*, pages 38–50. Springer.

Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., and Lakshminarayanan, B. (2021). Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64.

Pilipovic, P., Samson, A., and Ditlevsen, S. (2024). Parameter estimation in nonlinear multivariate stochastic differential equations based on splitting schemes. *The Annals of Statistics*, 52(2):842–867.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707.

Risken, H. (1989). Fokker-planck equation. In *The Fokker-Planck equation: methods of solution and applications*, pages 63–95. Springer.

Tang, K., Wan, X., and Liao, Q. (2022). Adaptive deep density approximation for fokker-planck equations. *Journal of Computational Physics*, 457:111080.