Detecting Stochastic Governing Laws with Observation on Stationary Distributions

Xiaoli Chen
National University of Singapore
Hui Wang (huiwang2018@zzu.edu.cn)
Zhengzhou University https://orcid.org/0000-0003-4574-8576
Jinqiao Duan
Illinois Institute of Technology

Research Article

Keywords: Stochastic dynamical systems, Fokker-Planck equations, Machine learning, Neural network.

Posted Date: June 16th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1549034/v1

Detecting Stochastic Governing Laws with Observation on Stationary Distributions *

Xiaoli Chen[1], Hui Wang[2*], Jinqiao Duan[3]

April 12, 2022

Abstract

Mathematical models for complex systems are often accompanied with uncertainties. The goal of this paper is to extract a stochastic differential equation governing model with observation on stationary probability distributions. We develop a neural network framework to learn the drift and diffusion terms of the stochastic differential equation. We introduce a new loss function containing the Hellinger distance between the observation data and the learned stationary probability density function. After minimizing this loss function in the training procedure, we find that the learned stochastic differential equation shows a reasonable approximation of the data-driven dynamical system. The effectiveness of our framework is demonstrated in numerical experiments.

Key words: Stochastic dynamical systems; Fokker-Planck equations; Machine learning; Neural network.

1 Introduction

Mathematical models for scientific and engineering systems often involve uncertainties and thus are often in the form of stochastic differential equations (SDEs). These stochastic dynamical systems are ubiquitous in biology, physics, geosciences and other fields. Stochastic dynamical systems provide an appropriate framework to investigate random phenomena [1–5]. Hence, the determination of SDE models are crucial for quantifying and predicting dynamical behaviors of nonlinear system under random fluctuations. An SDE is characterized by the drift (i.e., vector field) and diffusion terms. In this paper, we aim to detect appropriate drift and diffusion terms with stationary probability distribution data.

* [1] Department of Mathematics, National University of Singapore, 119077, Singapore. [2] School of Mathematics and Statistics, Zhengzhou University, Zhengzhou, China. [3] Department of Applied Mathematics & Department of Physics, Illinois Institute of Technology, Chicago 60616, USA. Correspondence and requests for materials should be addressed to H. Wang (Email: huiwang2018@zzu.edu.cn)

The stationary probability distribution of a stochastic dynamical system does not change with time and is the stationary solution of the corresponding Fokker-Planck equation [6–12]. The stationary probability distribution carries the information of the underlying stochastic system [13, 14].

The Hellinger distance is the distance between probability distributions that characterizes how close two different distributions are. In this paper, we use Hellinger distance to identify whether the constructed SDE is an appropriate approximation of a data-driven stochastic dynamical system.

Neural networks can be represented as compositions of simple functions with parameters, and such functional representations can be used for parameter estimation of time-series data and for kernel estimation [15]. There has been some progress in learning stochastic differential equation models from noisy data. A variation estimation method was used to learn the drift term with the observation trajectory data [16–19]. There was also an RNN-based variational method [20], a sparse learning method [21], and a Kramers-Moyal formulae [22] for learning stochastic dynamical systems. A stochastic adjoint sensitivity method was proposed to learn stochastic differential equations [23] or stochastic differential equations with jumps [24]. [25] used small samples from just a few snapshots of unpaired data to infer the drift and diffusion terms of stochastic differential equations. Moreover, [26, 27] learned Lévy noise parameters by deep neural networks. [28] solve the steady-state Fokker-Planck equation with a small amount of data through combining the deep KD-tree.

We have recently developed a data-driven approach [29] for discovering stochastic differential equations with non-Gaussian Lévy noise using the nonlocal Kramers-Moyal formulas, and further learned the stochastic differential equations from discrete particle samples at different time snapshots using the Fokker-Planck equation and physics-informed neural networks [30].

However, in addition to sample path observation data, there are recent advances in observing or measuring stationary probability distributions [7, 9, 11]. To take advantage of these new type of data, we devise a neural network framework to extract stochastic dynamical system models with stationary probability distribution as observation. This motivates our research reported in this paper. Specifically, we develop a neural network framework to extract stochastic governing laws based on probability measures. Given observation data of stationary probability density function, we learn the drift and diffusion terms which are approximated by two neural networks. If we learn the drift and diffusion together, the results would not be unique. We thus propose two methods. The first method is just to learn either the drift term or the diffusion term. The second method is to learn the drift and diffusion term together with one observation data of drift term. We compare our results with Hellinger distance replaced by Jensen-Shannon divergence and mean-square distance in three dimensional settings, which illustrate the effectiveness of our methods.

This paper is organized as follows. In Section 2, we present two methods for learning stochastic governing laws based on physics informed neural networks and Hellinger distance

of probability distributions. In Section 3, we present examples to learn the drift term and diffusion term. Finally, we end with some discussions in Section 4.

2 Methodology

2.1 Problem setup

Consider the following stochastic differential equation (SDE)

$$dX_t = b(X_t)dt + \sigma(X_t)dB_t, \quad X_0 = x_0, \tag{1}$$

where the n-dimensional vector function $b(\cdot)$ is the drift term, the $n \times n$ matrix function $\sigma(\cdot)$ is the diffusion term, and B_t is an n-dimensional Brownian motion.

The generator of the SDE (1) is [31]:

$$Au = \sum_{i=1}^{n} b_i \frac{\partial u}{\partial x_i} + \frac{1}{2} \sum_{i,j=1}^{n} (\sigma\sigma^T)_{i,j} \frac{\partial^2 u}{\partial x_i \partial x_j}.$$

The probability density function is a quantity that carries information of the stochastic system. The time evolving probability density function of the solution process X_t is governed by the Fokker-Planck equation, which is written as follows:

$$\begin{aligned} &\partial_t p(x,t) = A^* p(x,t), \qquad x \in \mathbb{R}^n, t > 0, \\ &p(x,0) = p_0(x), \end{aligned} \tag{2}$$

where $p_0(x)$ is the initial probability density function, A^* is the adjoint operator of the generator A and has the following form:

$$A^* p = -\sum_{i=1}^{n} \frac{\partial}{\partial x_i}(b_i p) + \frac{1}{2} \sum_{i,j=1}^{n} \frac{\partial^2}{\partial x_i \partial x_j}((\sigma\sigma^T)_{i,j} p). \tag{3}$$

Note that the Fokker-Planck equation is a deterministic linear partial differential equation with an initial condition.

The stationary Fokker-Planck equation is:

$$A^* p(x) = 0. \tag{4}$$

with a condition $\int_{R^n} p(x)dx = 1$. We assume that there exists a unique stationary probability density function (still denote it by $p(x)$) in this paper.

We consider the scenario that the available data is the probability density function. Our objective is to infer the drift and diffusion terms.

2.2 Theoretical foundation

As the drift b and diffusion σ characterize the uncertainty of the SDE, we will estimate them based on observations of probability distributions (i.e., probability measures) of the system paths X_t. Now we introduce the Hellinger distance [32, 33] between two probability distributions. It is used to quantify the distance between two probability distributions in the space of probability measures. For our purpose here, the Hellinger distance $H(p, q)$ between two probability density functions $p(x)$ and $q(x)$ is defined as follows,

$$H^2(p,q) \triangleq \frac{1}{2}\int_{\mathbb{R}} (\sqrt{p(x)} - \sqrt{q(x)}\ )^2 dx, \tag{5}$$

which satisfies the property: $0 \leq H(p, q) \leq 1$.

With the observed stationary probability density $q(x)$, we determine or estimate the vector field (i.e., drift) $b(x)$ by minimizing the Hellinger distance between the true stationary probability density $p(x)$ for the solution process $X(t)$ and the observation probability density $q(x)$.

Under appropriate conditions on b and σ (see [34, p.170]), such as, $b \leq 0$ and $\sigma \neq 0$ as well as some smoothness requirements, there exists a stationary probability density $p(x)$ for the SDE (1), as a solution of the steady Fokker-Planck equation,

$$p(x) = \frac{C}{\sigma^2(x)} e^{\int_{x^*}^{x} \frac{2b(y)}{\sigma^2(y)} dy}, \tag{6}$$

where the positive normalization constant C is chosen so that $p > 0$ and $\int_{\mathbb{R}} p(x)dx = 1$, i.e.,

$$C \triangleq 1/\int_{-\infty}^{\infty} \frac{e^{\int_{x^*}^{x} \frac{2b(y)}{\sigma^2(y)} dy}}{\sigma^2(x)}\, dx.$$

Note that x^* here may be an arbitrary reference point so that the integral $\int_{x^*}^{x} \frac{2b(y)}{\sigma^2(y)} dy$ exists. Different choice of x^* only affects the normalization constant C. (Say, take $x^* = 0$ if that is possible).

Given the observed stationary distribution q, we define the discrepancy function via Hellinger distance between the true stationary distribution $p(x)$ and $q(x)$. We denote

$$I(b,\sigma) \triangleq \frac{1}{2}\int_{\mathbb{R}} (\sqrt{p(x)} - \sqrt{q(x)}\ )^2 dx, \tag{7}$$

where $p(x)$ is in (6). The corresponding Euler-Lagrange equation is

$$\frac{d}{dt} I_\sigma = I_b. \tag{8}$$

Since the Euler-Lagrange equation is the necessary condition for the functional to obtain the minimum. So we can solve the corresponding Euler-Lagrange equation to get the minimum value of the functional.

Note that Hellinger distance is a measure to describe the distance of two probability density functions. Other distance also can describe it. Given the probability density function $p(x)$ and $q(x)$, respectively, the Kullback-Leibler (KL) divergence is defined as

$$H_{KL}(p||q) = \int p(x) \log(\frac{p(x)}{q(x)}) dx. \tag{9}$$

While the Kullback-Leibler divergence is asymmetry, another measure between two probability density Jensen-Shannon divergence is introduced as

$$H_{JS}(p||q) = \frac{1}{2} H_{KL}(p||\frac{q+p}{2}) + \frac{1}{2} H_{KL}(\frac{q+p}{2}||p). \tag{10}$$

Later on we will also use the Jensen-Shannon divergence to measure the distance.

2.3 Machine Learning

Given the noise intensity $\sigma(x)$ and observation of the stationary probability density $q(x)$, we will learn the drift term b. We devise two neural networks to approximate the drift term and stationary probability density $p(x)$, where the input is the space domain x and the output is the $b_{NN}(x)$ and $p_{NN}(x)$.

On the one hand, the output of $p_{NN}(x)$ should satisfy the functional (7). We define the loss function as:

$$Loss_H = \frac{1}{2} \sum_{i=1}^{N_H} (\sqrt{p_{NN}(x_i)} - \sqrt{q(x_i)})^2, \tag{11}$$

where $\{x_i\}_{i=1}^{N_H}$ are the points in the spatial domain to compute the integral and N_H is the number of the observation data.

On the other hand, the neural networks of the drift term $b_{NN}(x)$ and the stationary probability density $p_{NN}(x)$ should satisfy the steady Fokker-Planck equation:

$$A^* p = \partial_x \big(b(x)p(x)\big) + \frac{1}{2} \partial_{xx} \big(\sigma^2(x)p(x)\big) = 0.$$

Similar to the physics informed neural network [35, 36], we define the residual neural network as

$$f(x) = -\partial_x (b_{NN}(x) p_{NN}(x)) + \frac{\sigma^2}{2} \partial_{xx} p_{NN}(x). \tag{12}$$

Then the loss function of the residual neural network is defined as:

$$Loss_f = \frac{1}{N_f}\sum_{i=1}^{N_f}(f(x_i))^2, \tag{13}$$

where $\{x_i\}_{i=1}^{N_f}$ is the residual points in the spatial domain and N_f is the number of the residual points. Here we randomly choose the residual points at each iteration step. The sketch of the method is shown in Figure 1.

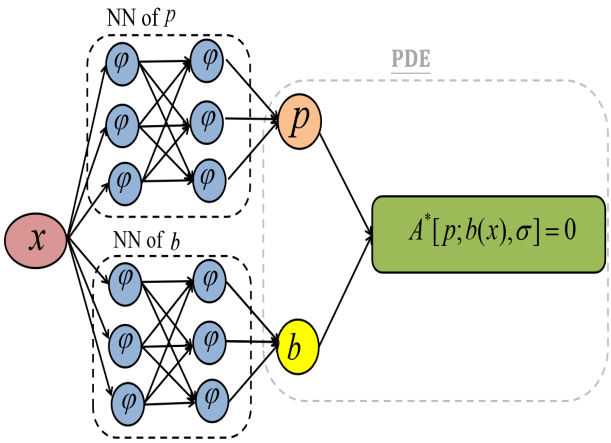

Figure 1: **Schematic of the neural network for solving PDEs**: two neural networks for approximate the probability p and the drift term b, where the input is x and φ is the activation function.

The total loss function is

$$Loss = Loss_H + Loss_f. \tag{14}$$

For the unknown drift term and diffusion term, because $b(x) = 0$ and $\sigma(x) = 0$ is also the minimization solution of loss function, thus we could not learn the terms uniquely if we train the loss function (14). The observation data [36] of drift term at some points need to know. To avoid the zeros solution, the observation data of drift term at few points is given, i.e. $\{x_i, b(x_i)\}_{i=1}^{N_b}$. The loss function of the drift term is:

$$Loss_b = \frac{1}{N_b}\sum_{i=1}^{N_b}(b_{NN}(x_i) - b(x_i))^2, \tag{15}$$

so the loss function is defined as:

$$Loss_2 = Loss_H + Loss_f + Loss_b. \tag{16}$$

If we use the Jensen-Shannon divergence to measure the distance, we just replace the $loss_H$ to

$$Loss_{JS} = H_{JS}(p_{NN}||q). \tag{17}$$

We also compare our method (14) with the traditional PINN loss [35] for solving the inverse problem of Fokker-Planck equation. With the observation data of the stationary probability density function $q(x)$, the loss function is written as mean square error:

$$Loss_{ob} = \frac{1}{N_{ob}} \sum_{i=1}^{N_{ob}} (p_{NN}(x_i) - q(x_i))^2. \tag{18}$$

The total loss of PINN method is defined as:

$$Loss_{PINN} = Loss_{ob} + Loss_f. \tag{19}$$

3 Numerical Experiments

The neural networks in our numerical experiments below have 4 hidden layers and 20 neurons per layer, with tanh activation function. The weights are initialized with truncated normal distributions. The biases are initialized as zero. We use the Adam optimizer with learning rate 10^{-4} to train the loss function.

3.1 Analytical estimation for the drift

Example 1. *Consider a stochastic model*

$$dX = b(X)dt + dB_t,$$

with function $b(x)$. Given an "observation" of the stationary probability density $q(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}$ (the Gaussian distribution). Find a function $b(x)$ so that the Hellinger distance $I(b(x)) = \frac{1}{2}\int_{\mathbb{R}}[\sqrt{p(x)} - \sqrt{q(x)}]^2 dx$ is minimized.

The true stationary probability density for the solution process X_t is

$$p(x) = \frac{e^{2\int_0^x b(y)dy}}{\int_{-\infty}^{\infty} e^{2\int_0^x b(y)dy} dx}.$$

The E-L equation is a necessary condition for functional minima.

$$I(b) = \frac{1}{2}\int_{\mathbb{R}} (p(x) + q(x) - 2\sqrt{p(x)}\sqrt{q(x)}\ )dx, \tag{20}$$

Submitting

$$p(x) = \frac{e^{2\int_0^x b(y)dy}}{\int_{-\infty}^{\infty} e^{2\int_0^x b(y)dy}dx}$$

and $q(x) = \frac{1}{\sqrt{2\pi}}\ e^{-\frac{1}{2}x^2}$ into (20) then

$$\begin{aligned} I(b) &= \frac{1}{2\sqrt{2\pi}}\int_{\mathbb{R}} e^{-\frac{1}{2}x^2}dx + \frac{1}{2}\int_{\mathbb{R}} \frac{e^{2\int_0^x b(y)dy}}{\int_{\mathbb{R}} e^{2\int_0^x b(y)dy}dx}dx - \int_{\mathbb{R}} \sqrt{\frac{e^{-\frac{1}{2}x^2+2\int_0^x b(y)dy}}{\sqrt{2\pi}\int_{\mathbb{R}} e^{2\int_0^x b(y)dy}dx}}dx \\ &:= I_1 + I_2 + I_3 \end{aligned} \tag{21}$$

So , we need $I_b = I_{1b} + I_{2b} + I_{3b} = 0$. By calculating, we get $b(x) = -kx, k \geq 0$. Submitting $b(x)$ into $p(x)$, then $p(x) = \sqrt{\frac{k}{\pi}}e^{-kx^2}$, which satisfies $\int_{-\infty}^{\infty} p(x)dx = 1$. The error $Err = \|p(x) - q(x)\|_H$, submitting $p(x)$ and $q(x)$ into Err:

$$\begin{aligned} Err =& \|p(x) - q(x)\|_H = \frac{1}{2}\int_R (\sqrt{p(x)} - \sqrt{q(x)})^2 dx \\ =& \frac{1}{4\sqrt{\pi}} + \sqrt{\frac{k}{2\pi}} - \sqrt{\frac{4k}{2\pi(k+\frac{1}{2})}} \end{aligned} \tag{22}$$

This is a function $f(k)$ about variable k. f_{min} attains when $k = \frac{1}{2}$. So, $b(x) = -\frac{1}{2}x$.

3.2 Machine learning the drift and diffusion

Example 2. *Consider a scalar stochastic model*

$$dX = b(X)dt + \sigma dB_t,$$

with drift function $b(x) = x - x^3$. Given an "observation" of the stationary probability density $q(x) = \frac{1}{A}e^{\frac{1}{\sigma^2}x^2 - \frac{x^4}{2}}$, where $A = \int_{\mathbb{R}} e^{\frac{1}{\sigma^2}x^2 - \frac{x^4}{2}}dx$. Find a drift function $b(x)$ so that the Hellinger distance $I(b(x)) = \frac{1}{2}\int_{\mathbb{R}}[\sqrt{p(x)} - \sqrt{q(x)}]^2 dx$ is minimized.

Given the noise intensity (diffusion) $\sigma = 1$, we use two fully connected neural networks to approximate the drift term and stationary probability density respectively. We choose

$N_H = 1001$, $N_f = 10000$ to train the loss function (14). The results we learned is shown in Figure 2. In Figure 2 (a), we plot the true drift term (black line) and the learned drift term (red line). The neural network can approximate the drift very well. In Figure 2 (b), the given $q(x)$ and neural network result of $p_{NN}(x)$ can approximate well too. We also plot the loss function evolves with the number of iterative steps. The loss is less than 10^{-4}.

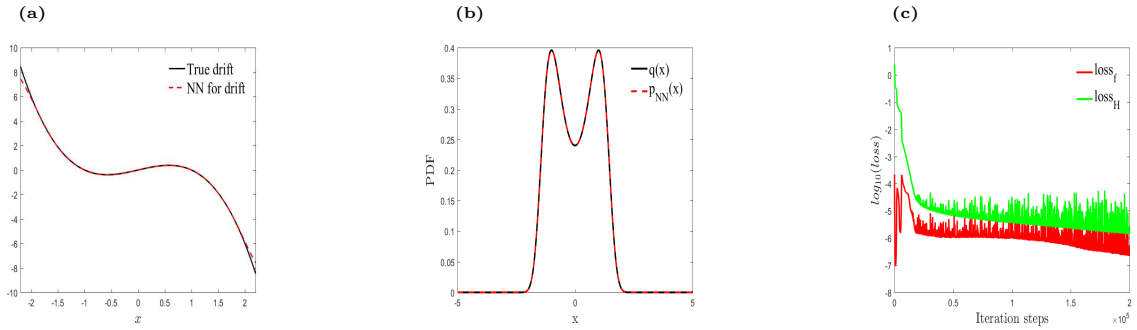

Figure 2: **Unknown drift of example 2.** (a) The learned drift term; (b) The learned probability; (c) The loss function.

For the unknown drift term and diffusion term, we train the loss function (16) to get the optimal results. Only one observation data of drift term at $x = -2$ is given. The results are shown in Figure 3. We present learned drift result in Figure 3 (a), and the diffusion term evolution predictions as the iteration of the optimiser progresses in Figure 3 (b). The neural network of probability density is shown in Figure 3 (c). We can see for one observation data of the drift term, the drift and diffusion term can be learned well. When $|x| > 2$, the error of learned drift term becomes larger than that for the other x. Because the probability in larger x is almost zero, we do not know much information there.

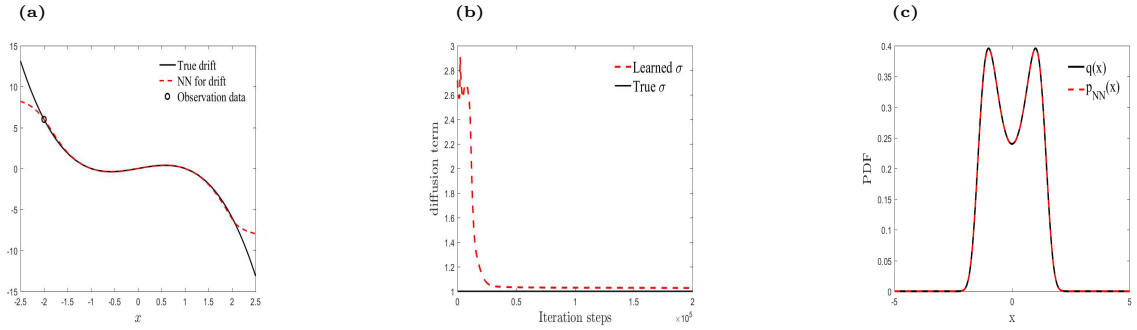

Figure 3: **Unknown drift and diffusion of Example 2.** (a) the learned drift term with 1 observation data at $x = -2$; (b) the learned diffusion term; (c)the learned probability density.

Example 3. *Consider a scalar stochastic model*

$$dX = b(X)dt + \sigma dB_t,$$

with drift function $b(x)$. Given an "observation" of the stationary probability density $q(x) = \frac{1}{\pi}\frac{1}{1+x^2}$. Find a function $b(x)$ so that the Hellinger distance $I(b(x)) = \frac{1}{2}\int_{\mathbb{R}}[\sqrt{p(x)}-\sqrt{q(x)}]^2dx$ is minimized.

Similar to the last example, we fix the noise intensity (diffusion) $\sigma = 1$, and use two neural networks to approximate the drift term and stationary probability density respectively. Here we also choose $N_H = 1001$, $N_f = 10000$.

$$I(b(x)) = \frac{1}{2}\int_{\mathbb{R}}(\sqrt{p(x)} - \sqrt{q(x)})^2dx, \tag{23}$$

where $q(x) = \frac{1}{\pi}\frac{1}{1+x^2}$.

The results are shown in Figure 4. In this case, the true drift term $b(x)$ is unknown, so we could not compare the drift term. By comparing the learned stationary distribution and the observation data, the result shows that they match well. What is more, the loss function is sufficiently small, and the probability density distribution is concentrated around zero. So zero could be the stable point of this system. Our learned drift term $b(x)$ has one stable point zero, as in Figure 4. This is in line with our expectations.

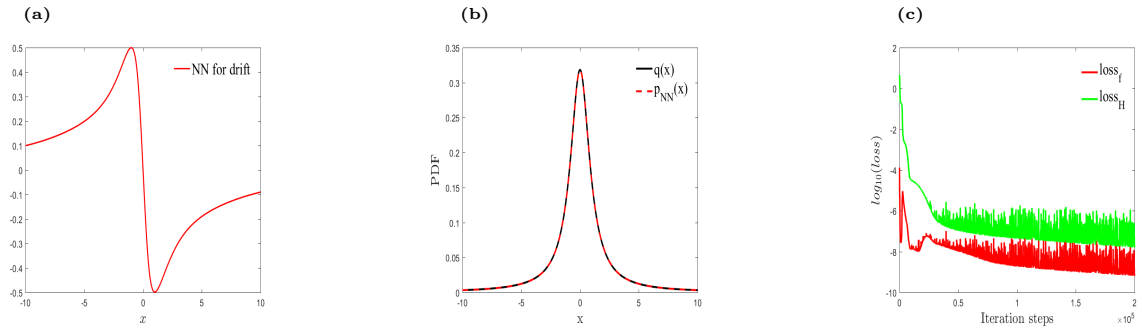

Figure 4: **Unknown drift of Example 3.** (a) The learned drift term; (b) The learned probability; (c) The loss function of OM part and the equation.

Molecular and cell biology is playing an increasingly important role in life sciences.For example, a number of research findings are about stochastic fluctuations inducing phenotypic diversity in gene expression. Here we consider a stochastic gene regulation model [37, 38].

Example 4. *This is a stochastic model for a transcription factor (i.e., a protein) concentration evolution in a certain gene regulation network*

$$dX = b(X)dt + \sigma dB_t,$$

with drift function $b(x) = \frac{k_f x^2}{x^2+K_d} - k_d x + R_{bas}$. Here the parameters are $K_d = 10$, $k_d = 1min^{-1}$, $k_f = 6min^{-1}$, and $R_{bas} = 0.4min^{-1}$. We will find a drift function $b(x)$ and diffusion σ so that the Hellinger distance $I(b(x), \sigma) = \frac{1}{2}\int_{\mathbb{R}}[\sqrt{p(x)} - \sqrt{q(x)}]^2dx$ is minimized.

Given the noise intensity $\sigma = 1$, two neural networks are used to approximate the drift term and stationary probability density respectively. Here we still choose $N_H = 1001$, $N_f = 10000$. The result of the drift is shown in Figure 5 (a), where the black line is the true drift term and red line is the neural network result. Obviously, we find that he learned drift term can fit the true result very well. And the learned probability density function is shown in Figure 5 (b), which fits very well with the the observation probability $q(x)$. We also show the loss function of $loss_H$ and $loss_f$ in Figure 5 (c). We see that the loss function decreases fast with the iteration steps increasing.

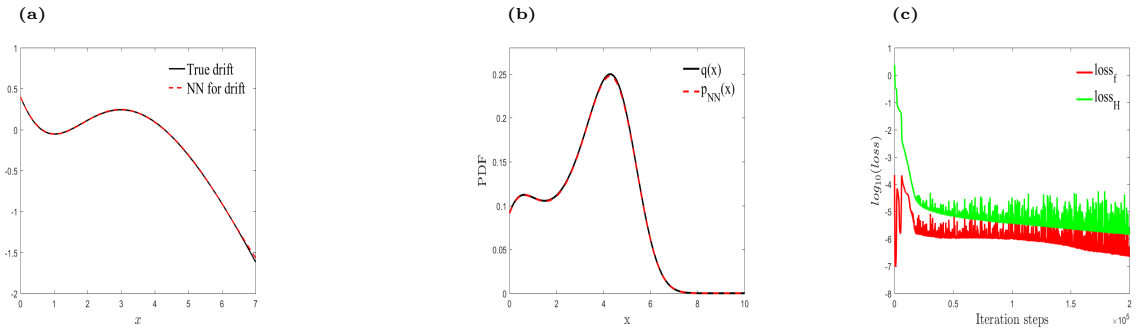

Figure 5: **Unknown drift of Example 4.** (a) The learned drift term; (b) The learned probability; (c) The loss function.

For the unknown drift term and diffusion term case, we train the loss function (16) to get the optimal result. Only one observation data of drift term at $x = 5$ is given. The results are shown in Figure 6. We present learned drift result in Figure 6 (a), and the diffusion term evolution predictions as the iteration of the optimiser progresses in Figure 6 (b). The neural network result of probability density is shown in Figure 6 (c). From the figures, we see that the drift and diffusion term can be learned well.

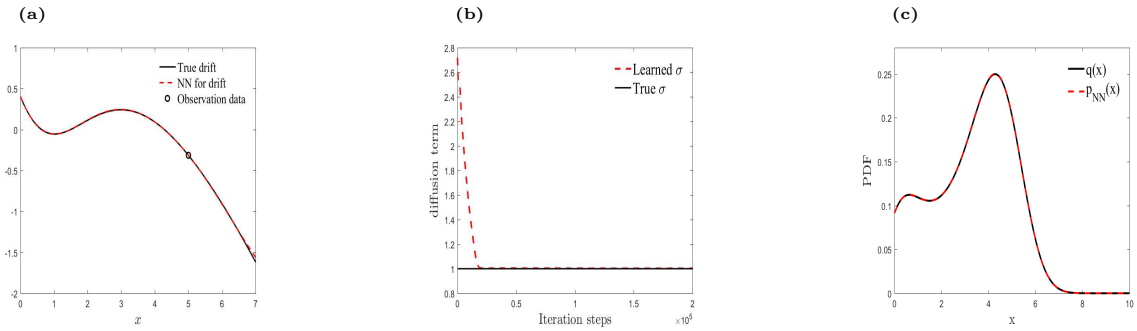

Figure 6: **Unknown drift of example 4.** (a) the learned drift term with 1 observation data at $x = 5$; (b) the learned diffusion term; (c) the learned probability density function.

Example 5. *We now consider the following three dimensional stochastic dynamical systems*

with non-polynomial drift [30]:

$$d\begin{pmatrix} X_t \\ Y_t \\ Z_t \end{pmatrix} = \begin{pmatrix} -\partial_{X_t}\Phi(X_t,Y_t,Z_t) \\ -\partial_{Y_t}\Phi(X_t,Y_t,Z_t) \\ -\partial_{Z_t}\Phi(X_t,Y_t,Z_t) \end{pmatrix} dt + \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \end{bmatrix} d\begin{pmatrix} B_{1,t} \\ B_{2,t} \\ B_{3,t} \end{pmatrix}, \tag{24}$$

where the potential $\Phi(x,y,z) = -\frac{1}{2}\log[(2\exp(\lambda_{01}(x-\lambda_{11})^2 + \lambda_{02}(y-\lambda_{12})^2 + \lambda_{03}(z-\lambda_{13})^2) + \exp(\lambda_{04}(x-\lambda_{14})^2 + \lambda_{05}(y-\lambda_{15})^2 + \lambda_{06}(z-\lambda_{16})^2)]$, $\lambda_{0i} = -5,-2.5,-5,-1,-1,-1$, $\lambda_{1i} = 1,1,1,-2,-1,-1$ *and* $\sigma_j = 1$, *where* $i = 1,2,...,6$ *and* $j = 1,2,3$. *The "observation" of the stationary probability density is* $q(x,y,z) = 1/Z\exp(-2\Phi(x,y,z))$, *where* Z *is the normalization parameter such that the integral of* $q(x,y,z)$ *on domain* R^3 *is equal to 1. Find the parameters in drift term and diffusion term so that the Hellinger distance* $I = \frac{1}{2}\int_{\mathbb{R}^3}[\sqrt{p(x,y,z)} - \sqrt{q(x,y,z)}]^2 dxdydz$ *is minimized.*

We use neural network to approximate the stationary probability density. And here we choose $N_H = 50,000$, $N_f = 5000$.

First, we learn all the parameters λ_{0i}, λ_{1i} and σ_j, for $i = 1,2,...6$ and $j = 1,2,3$. The results are shown in Figure 7. In Figure 7(a) and (c), the parameters of λ_{0i} and drift term are learned not well. While the parameters λ_{2i} can be learned well, see Figure 7 (b). So we will learn the parameter in the drift term and diffusion term respectively.

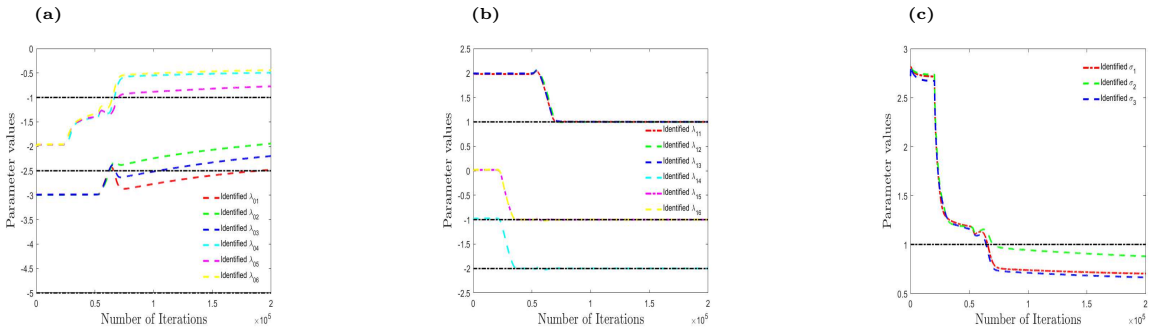

Figure 7: **Three dimensional results.** Learn all the parameter in the SDE. (a) learned λ_{0i}; (b) learned λ_{1i}; (c) learned σ_j, where $i = 1,2,...,6$ and $j = 1,2,3$.

On the one hand, we just learn the diffusion term given the drift term. The results are shown in Figure 8. The parameters in diffusion term approaches to the true parameter as the number of iterations increases.

On the other hand, we learn the parameters λ_{0i} and λ_{1i} in the drift term given the diffusion term. The results are shown in Figure 9. We learn the results for three cases. For case I: the observation data is clean, i.e. $q(x,y,z)$. For case II: the observation data is given with 5% noise, i.e. $q(x,y,z)*(1+0.05\mathrm{N}(0,I))$, and for case III: the observation data is given with 10% noise, i.e. $q(x,y,z)*(1+0.1\mathrm{N}(0,I))$. Here $\mathrm{N}(0,I)$ mean the standard normal distribution. The parameters we learned are well even the observation data has 10% noise.

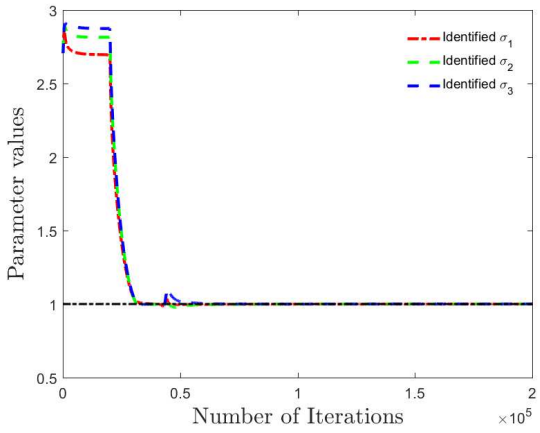

Figure 8: **3D result** the learned drift term.

In the following, We change the Hellinger distance to be the Jensen-Shannon divergence in the loss function. The results of learned parameters in the drift term are shown in Figure (10). The unknown parameters can be learned well. While comparing with the Hellinger distance, this method need more iteration step to train.

Here we also compare our results with the traditional physics informed neural network (PINN) with the case of learning the parameter in the drift term. The results are shown in Figure 11. Only several parameters in the drift term can be learned well using PINN method. Compared with PINN method using mean square error (19), our loss with Hellinger distance (14) would get better results. We use the neural network to approximate the probability density function and plot the learned PDF when $z = -1, 0.5, 1$ using different methods. The results are shown in Figure 12. Comparing with the true probability density function, the proposed method with Hillinger distance and Jensen-Shannon divergence works better than the mean square distance.

4 Discussion

In summary, we have devised a data-driven framework to extract stochastic dynamical systems models from observation data on stationary probability distributions. This framework is based on minimizing Hellinger distance between two probability distributions. Our numerical results in one and three dimensional examples have verified that this framework is feasible. We have also compared with Jensen-Shannon distance, and it has turned out that our framework also works with other distances in the space of probability distributions.

This approach leads to a data-driven stochastic dynamical systems study of random phenomena, as we can further examine dynamical behaviors of the leaned stochastic governing laws [31].

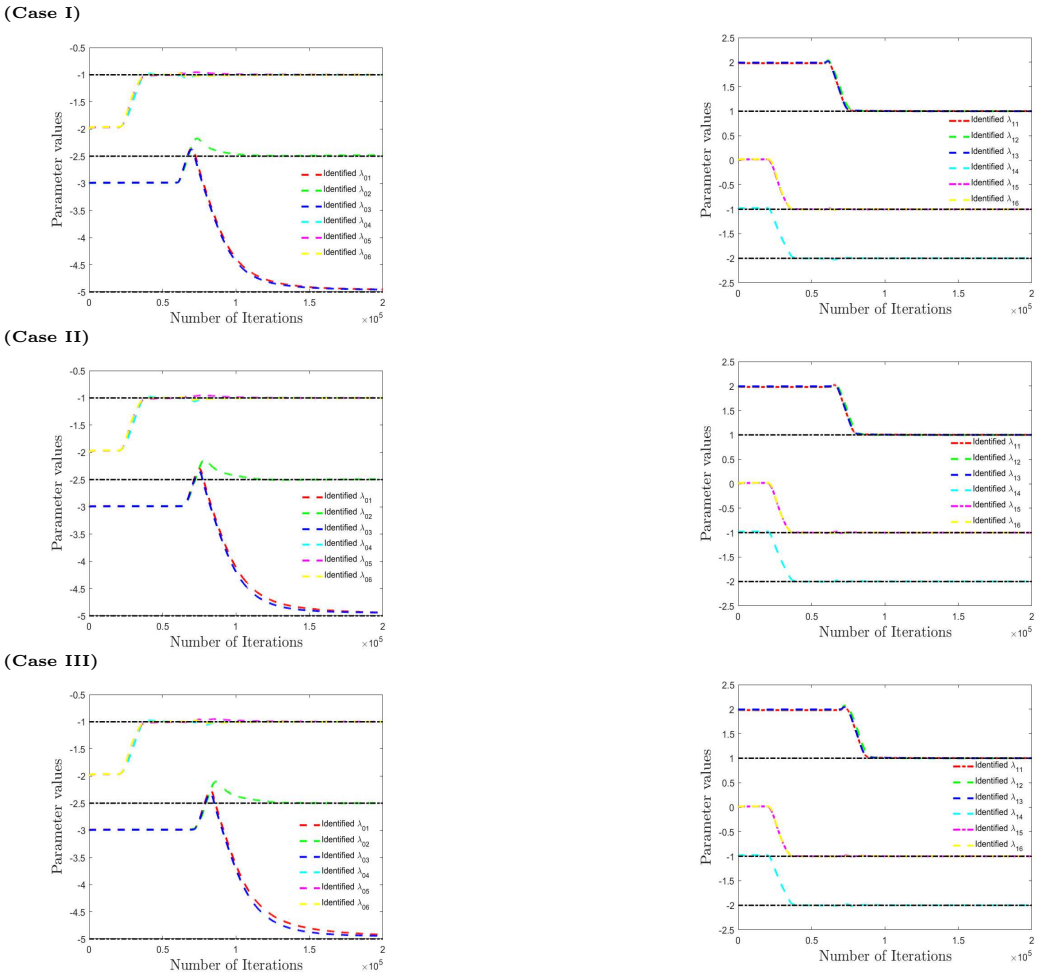

Figure 9: **Three dimensional results.** Learn all parameters in the drift terms. Case I: clean observation data of the PDF; Case II: 5% noise observation data of the PDF; Case III: 10% noise observation data of the PDF. Left: learned λ_{0i}; right: learned λ_{1i}, where $i = 1, 2, ..., 6$ and $j = 1, 2, 3$.

Acknowledgements

We would like to thank Xi Chen for helpful discussions. This work is supported by the National Natural Science Foundation of China (NSFC) (Grant No.11901536).

Data availability

Data sharing is not applicable to this article as no data sets were generated or analyzed during the current study.

Figure 10: **Three dimensional results.** Learn parameters in the drift terms using Jensen-Shannon distance. Left: learned λ_{0i}; right: learned λ_{1i}, where $i = 1, 2, ..., 6$ and $j = 1, 2, 3$.

Figure 11: **Three dimensional results with PINN loss.** Learn all the parameter in the drift term with clean observation data of the PDF. Left: learned λ_{0i}; right: learned λ_{1i}, where $i = 1, 2, ..., 6$ and $j = 1, 2, 3$.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

References

[1] Arnold, L.: *Random Dynamical Systems*. New York, Springer, Corrected 2nd printing(2003)

[2] Pasquero, C., Tziperman, E.: Statistical parameterization of heterogeneous oceanic convection, *Journal of Physical Oceanography*. 37: 214-229(2007)

[3] Penland, C., Sura, P.: Sensitivity of an ocean model to "details" of stochastic forcing. In *Proc. ECMWF Workshop on Represenation of Subscale Processes using Stochastic-Dynamic Models*. Reading, England, 6-8 June(2005)

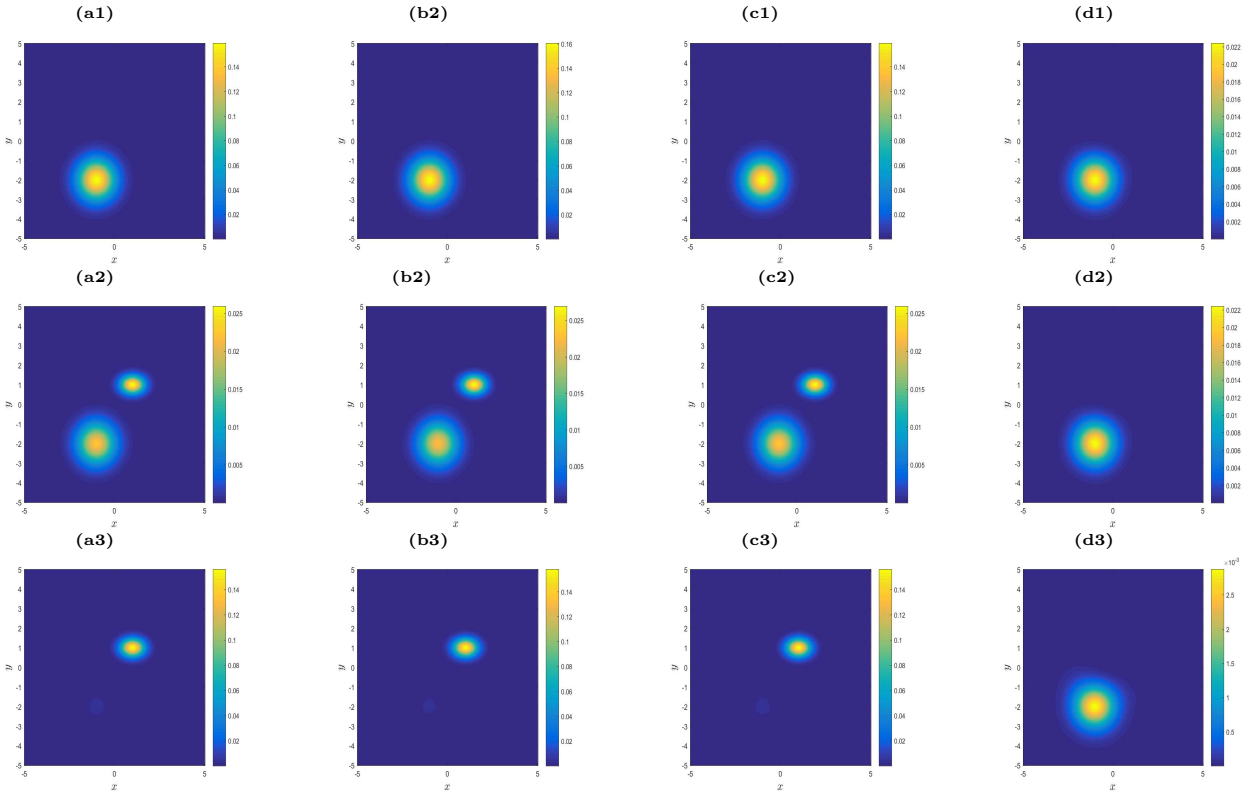

Figure 12: **The probability density function of Example 5.** (a1)-(a3): The true PDF with $z = -1, 0.5, 1$; (b2)-(b3): the learned PDF using Hellinger distance (14); (c1)-(c3): the learned PDF using Jensen-Shannon divergence loss (17); (d1)-(d3): the learned PDF using PINN loss (19).

[4] Gao, T., Duan, J.: Quantifying model uncertainty in dynamical systems driven by non-gaussian levy stable noise with observations on mean exit time or escape probability. Communications in Nonlinear Science and Numerical Simulation. 39: 1-6(2016)

[5] Wu, D., Fu, M., Duan, J.: Discovering mean residence time and escape probability from data of stochastic dynamical systems. Chaos. 29(9):093122(2019)

[6] Hung, C. L., Zhang, X., Gemelke, N., Chin, C.: Observation of scale invariance and universality in two-dimensional Bose gases. Nature. 470(7333), 236-239(2011)

[7] Hairapetian, G., Stenzel, R.: Observation of a stationary, current-free double layer in a plasma. Physical Review Letters. 65(2):175(1990)

[8] Yarmchuk, E. J., Gordon, M. J. V., Packard, R. E.: Observation of Stationary Vortex Arrays in Rotating Superfluid Helium. Physical Review Letters. 43(3):214-217(1979)

[9] Gefen, O., Fridman, O., Ronin, I., Balaban, N, Q.: Direct observation of single stationary-phase bacteria reveals a surprisingly long period of constant protein production activity. Proceedings of the National Academy of Sciences. 111(1):556-561(2014)

[10] Arnold, L., Wishtutz, V.: Stationary solutions of linear systems with additive and multiplicative noise. Stochastics-an International Journal of Probability & Stochastic Processes. 7(1-2):133-155(1982)

[11] Liberzon, D., Brockett, R. W.: Nonlinear feedback systems perturbed by noise: steady-state probability distributions and optimal control. Automatic Control IEEE Transactions on. 45(6):1116-1130(2000)

[12] Gray, A. H.: Uniqueness of Steady-State Solutions to the Fokker-Planck Equation. Journal of Mathematical Physics. 6(4):644-647(1965)

[13] Khasminskii, R.: Stochastic stability of differential equations, Springer(2012)

[14] Schmalfuss, B.: Lyapunov functions and non-trivial stationary solutions of stochastic differential equations. Dynamical Systems: An International Journal. 16(4), 303-317(2001)

[15] Gerber, F., Nychka, D. W.: Fast covariance parameter estimation of spatial Gaussian process models using neural networks. *Statistics and Probability*. 10(1)(2021)

[16] Batz, P., Ruttor, A., Opper, M.: Variational estimation of the drift for stochastic differential equations from the empirical density. Journal of Statistical Mechanics- Theory and Experiment. (8), 083404(2016)

[17] Batz, P., Ruttor, A., Opper, M.: Approximate Bayes learning of stochastic differential equations. Physical Review E. 98(2), 022109(2018)

[18] Opper, M.: An estimator for the relative entropy rate of path measures for stochastic differential equations. Journal of Computational Physics. 330, 127-133(2017)

[19] Opper, M.: Variational inference for stochastic differential equations. Annals of Physics. 531(3), 1800233(2019)

[20] Ryder, T., Golightly, A., McGough, A. S., Prangle, D.: Black-box variational inference for stochastic differential equations. ICML. 4423-4432(2018)

[21] Boninsegna, L., Nüske, F., Clementi, C.: Sparse learning of stochastic dynamical equations. Journal of chemical physics. 148(24), 241723(2018)

[22] Tabar, R.: Analysis and data-based reconstruction of complex nonlinear dynamical systems, Berlin/Heidelberg, Germany: Springer(2019)

[23] Li, X., Wong, T. K. L., Chen, R. T., Duvenaud, D.: Scalable gradients for stochastic differential equations. AISTATS. 3870-3882(2020)

[24] Jia, J., Benson, A. R.: Neural jump stochastic differential equations, NIPS. 32(2019)

[25] Yang, L., Daskalakis, C., Karniadakis, G. E.: Generative Ensemble Regression: Learning Particle Dynamics from Observations of Ensembles with Physics-Informed Deep Generative Models. SIAM Journal on Scientific Computing. 44(1), B80-B99(2022)

[26] Zhang, H., Xu, Y., Li, Y., Kurths, J.: Statistical solution to SDEs with α-stable Lévy noise via deep neural network. International Journal of Dynamics and Control. 8(4), 1129-1140(2020)

[27] Xu, Y., Zhang, H., Li, Y., Zhou, K., Liu, Q., Kurths, J.: Solving Fokker-Planck equation using deep learning. Chaos. 30(1), 013133(2020)

[28] Zhang, H., Xu, Y., Liu, Q., Wang, X., Li, Y. Solving fokkerCplanck equations using deep kd-tree with a small amount of data. Nonlinear Dynamics. (2022) https://doi.org/10.1007/s11071-022-07361-2.

[29] Li, Y., Duan, J.: A data-driven approach for discovering stochastic dynamical systems with non-Gaussian Lévy noise. Physica D. 417, 132830(2021)

[30] Chen, X., Yang, L., Duan, J., Karniadakis, G. E.: Solving Inverse Stochastic Problems from Discrete Particle Observations Using the Fokker-Planck Equation and Physics-Informed Neural Networks. SIAM Journal on Scientific Computing. 43(3) B811-30(2021)

[31] Duan, J.: *An Introduction to Stochastic Dynamics*, New York: Cambridge University Press(2015)

[32] Cha, S.: Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions. *International Journal of Mathematical Models and Methods in Applied Sciences*. 1(4): 300-307(2007)

[33] Beran, R.: Minimum hellinger distance estimates for parametric models. Annals of Statistics. 5(3):445-463(1977)

[34] Klebaner, F. C.: *Introduction to Stochastic Calculus with Applications*. Imperial College Press, London, 2nd edition(2005)

[35] Raissi, M., Perdikaris, P., Karniadakis, G. E.: 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(2019)

[36] Chen, X., Duan, J., Karniadakis, G. E.: Learning and meta-learning of stochastic advection-diffusion-reaction systems from sparse measurements. European Journal of Applied Mathematics. 32(3):397-420(2020)

[37] Wang, H., Cheng, X., Duan, J., Kurths, J., Li, X.: Likelihood for transcriptions in a genetic regulatory system under asymmetric stable Levy noise. Chaos. 28, 013121(2018)

[38] Cheng, X., Wang, H., Wang, X., Duan, J., Li, X.: Most probable transition pathways and maximal likely trajectories in a genetic regulatory system. Physica A. 531, 121779(2019)