

論文紹介

Conditioning Diffusions Using Malliavin Calculus

榎本 拓実

August 31, 2025

著者: Jakiw Pidstrigach et.al. ([1], ICML2025)

- ・ 確率制御や条件付き生成モデルでは終端時刻に対する評価関数を最大化するよう拡散プロセスを更新するタスクが重要
- ・ 実際の問題設定では評価関数が特異になることが多い
- ・ Malliavin 微分の活用により特異な評価関数をロバストに取り扱える手法を開発

以下 SDE に従う潜在変数を考える:

$$dX_t = b_t(X_t)dt + \sigma_t(X_t)dB_t, X_0 = x_0$$

観測変数 $Y = G(X_T)$

$Y = y$ の条件の下で X_t をサンプリングしたい

コントロール変数 u_t を導入し, 良い u_t を選んで SDE シミュレーションにより効率的にサンプリングできるようにしたい:

$$dX_t = b_t(X_t)dt + u_t(X_t)dt + \sigma_t(X_t)dB_t$$

G は尤度関数を誘導:

$$g(x; y) := p(Y = y | X_T = x).$$

g は多くの場合に singular. G が smooth, deterministic でも singular となりうる

ex. G が indicator function or 分類問題の時のような不連続関数を挟む場合. 特に, 正解ラベルで条件付ける等

ex. ("Diffusion Bridge") $G = \text{Id}$ でも $g(x; y) = \delta_y(x)$.

- Doob h -transform により考えたい条件付き分布を再現する最適コントロール表現 u^* を与える
- u^* はスコア関数を含む \rightarrow スコア関数を一般化 Tweedie 公式により条件付き期待値表現
 - Tweedie 公式の証明で Malliavin 微分 + 部分積分を利用
- 条件付き期待値 = Hilbert 空間での projection とみなし, ノルム差最小化として数値的に求める
- SDE を学習するのではなく, 与えられた前提で考える手法

Jacobian Process

SDE の解を初期値に関して微分可能であり, かつその微分値は誘導される SDE に従う

Theorem (Ito-Watanabe etc.)

係数が微分可能ならば SDE の解は空間方向に微分可能な修正をもち, その微分値は

$$dJ_t = \nabla b_t(X_t)J_t dt + \nabla \sigma_t(X_t)J_t dB_t$$

に従う

Remark

定式化に *Jacobian process* が必要だが, 直接これを計算には利用しない (Algorithm2. ある種の *back propagation*)

Main theorem

$\{B_t\}$: Brownian motion on (Ω, \mathcal{F}, P)

$Y = G(X_T, \xi)$, ξ と \mathcal{F}_T は独立 (観測時のノイズも考慮)

Theorem ([1], Theorem 2.1)

$\alpha : [0, T] \rightarrow \mathbb{R}^{n \times n}$: C^∞

$A_{T|s} := \alpha_T - \alpha_s$: invertible for all $s \in [0, T]$.

$\mathcal{S}_s := A_{T|s}^{-1} \int_s^T \alpha'_t J_{t|s}^\top (\sigma_t(X_t)^\top)^{-1} dB_t$

$\mathcal{L}(u) := \mathbb{E}^P \left[\int_0^T \|u_s(X_s; Y) - \mathcal{S}_s\|^2 ds \right]$

この時, \mathcal{L} は唯一つの最小解 u^* をもち,

$$dX_t = b_t(X_t)dt + \sigma_t(X_t) \sigma_t(X_t)^\top u_t^*(X_t; y)dt + \sigma_t(X_t)dB_t$$

の分布は元の SDE の解に対しての $Y = y$ の元での条件付き分布と一致.

Main theorem を用いて, 以下アルゴリズムを提案

(BEL = Bismut-Elworthy-Li)

- ・ コントロール変数 u_t (=スコア S_t の条件付き期待値) を NN でモデル化
- ・ B, X, Y を (条件付けず) サンプルング. $u_t(X_t, Y)$ を得る
- ・ パス (B, X) から $S_t(X, B)$ を計算
- ・ これを繰り返し誤差関数をモンテカルロ積分
- ・ u_t のパラメータをモンテカルロ積分の勾配により更新

Remark

α はある種のハイパーパラメータ.

学習対象の S_t の期待値は一致するが、分散が異なる.

分散が小さければ学習に利用する勾配の分散も小さくなる

Algorithm 1 BEL - Training Step

Require: $\alpha : [0, 1] \rightarrow \mathbb{R}^{n \times n}$, initial condition x_0 , batch size N , current drift approximation u^θ , time grid $\{t_0, t_1, \dots, t_M\}$.

- 1: **for** $i = 1$ to N **do**
- 2: Sample a sample path X with corresponding Brownian motion path B from the SDE (6).
- 3: Sample an observation $Y = G(X_T)$ from (2).
- 4: Compute the Monte Carlo estimator for \mathcal{S}_s using (8) (for details see Algorithm 2) along the path (X, B) .
- 5: Calculate the single-path loss

$$l_i(\theta) = \sum_{j=1}^{M-1} \|u_{t_j}^\theta(X_{t_j}; Y) - \mathcal{S}_{t_j}(X, B)\|^2,$$

- 6: **end for**
- 7: Sum for the full-batch loss

$$\mathcal{L}^N(\theta) = \sum_{i=1}^N l_i(\theta).$$

- 8: Take a gradient step on $\mathcal{L}^N(\theta)$ with your favourite optimiser.
-

Proposition

システムの SDE の解は十分に滑らかな確率密度関数を持つと仮定.

$y \in \mathbb{R}^d$ に対し

$$u_t(x; y) := \sigma_t(x) \sigma_t(x)^\top \nabla_x \log p_{T|t}(Y = y | X_t = x)$$

と定める.

この時, コントロール含めた SDE の解の分布は元のシステムの SDE の解の $Y = y$ の下での条件付き分布と一致する

数値計算上, これを直接使えるか?? \rightarrow NO...

$\therefore p_{T|t}$ の計算が困難

Generalized Tweedie formula

Conditional score の計算に Tweedie formula (Efron) が重要

密度関数, スコア: 計算難

→ 期待値で表現することで Monte Carlo が利用できる!

Proposition ([1] Prop.2.3)

Main theorem の仮定の下, $t \in [0, T)$, $x \in \mathbb{R}^n$, $y \in \mathbb{R}^d$ に対し

$$\nabla_x \log p_{T|t}(Y = y | X_t = x) = \mathbb{E}[\mathcal{S}_t | X_t = x, Y = y]$$

この証明に Malliavin 解析を利用する

Generalization of Bismut-Elworthy-Li formula

Theorem ([1] Thm.D.1)

$\varphi \in C_c(\mathbb{R}^n)$: test function (急減少関数でも良い)

$\alpha : [0, T] \rightarrow \mathbb{R}^{n \times n}$, matrix-values, differentiable

$A_{T|s} := \alpha_T - \alpha_s$: invertible for $\forall s \in [0, T)$.

For $T > s$, it holds that

$$\begin{aligned} \nabla_{x_s} \mathbb{E}[\varphi(X_T) | X_s = x_s] = \\ \mathbb{E} \left[\varphi(X_T) \int_s^T (\sigma_t(X_t)^{-1} J_{t|s} \alpha'_t)^\top dB_t | X_s = x_s \right] A_{T|s}^{-1} \end{aligned}$$

Remark

Watanabe[2] の理論により, φ は超関数に拡張できる.

$\varphi = \delta_{x_T}$ として $p_{T|s}(x_T | x_s) = \mathbb{E}[\delta_{x_T}(X_T) | X_s = x_s]$ を使う.

Generalization of Bismut-Elworthy-Li formula

Proof.

Malliavin 微分の chain rule より

$$D_s\varphi(X_T) = \nabla\varphi(X_T)D_sX_T = \nabla\varphi(X_T)J_{T|s}\sigma_s(X_s).$$

これと通常の chain rule

$$\nabla_x[\varphi(X_T^x)] = \nabla\varphi(X_T)J_{T|0}.$$

を合わせ, 両辺を $\alpha'_s ds$ 上で積分すれば

$$\nabla_x[\varphi(X_T^x)](\alpha_T - \alpha_0) = \int_0^T D_s\varphi(X_T)\sigma_s(X_s)^{-1}J_{s|0}\alpha'_s ds.$$

両辺期待値を取り, あとは部分積分公式.

□

Proof.

条件付き期待値は Hilbert 空間 $L^2(\Omega)$ での閉部分空間への射影でもあるから,

$$\mathcal{L}_{\text{local}}^t(u) := \mathbb{E} [\|u_t(X_t, Y) - \mathcal{S}_t\|^2]$$

とおいた時

$$u^* = \arg \min \mathcal{L}_{\text{local}}^t(u) = \mathbb{E}[\mathcal{S}_t | X_t, Y] = \nabla_x \log p_{T|t}(Y = y | X_t = x).$$

これと Doob h -transform を組み合わせればよい.

□

論文での実験例

- Brownian Bridge での解析解との比較
- Double well problem
- Shape process
- Fashion-MNIST でのデータ生成

数理生物の問題: 細胞や器官の形態変化のモデリング

$$Y = \{y^j\}_{j=1}^M \subset \mathbb{R}^2,$$

$\{B^j\}$: 互いに独立な 2-dim BM

点群 $X_t = \{x_t^i\}_{i=1}^N \subset \mathbb{R}^2$ の時間発展を以下で与える:

$$dx_t^i = \sum_{j=1}^M k(y^j, x_t^i) dB_t^j, 1 \leq i \leq N$$

$$k(x, y) = \kappa \frac{\|x - y\|_2^2}{\beta}$$

$$M = N, \kappa = 0.1, \beta = 1.0$$

初期値 X_0 : 半径 1 の円上 $\rightarrow X_T = Y$: 半径 1.5 の円上で条件付け

Shape Process での実験

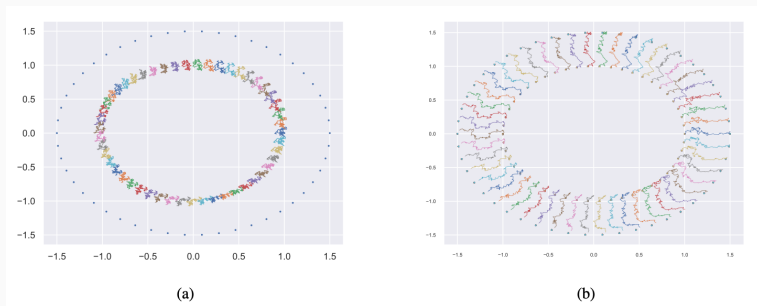


Figure 1: 出典: [1] Figure 8

(a) 条件なしでデータ生成

(b) BEL average ($\alpha_s := s$)

過去の研究例と精度を比較:

- Baker et al. (2025): Feymann-Kac ベースの”adjoint path”
- Heng et al. (2022): time-reversal

Method	Dist
BEL average	0.085
Time reversal	0.090
Adjoint paths	0.498
Untrained	1.396

Figure 2: 出典: [1] Table 1

Time-reversal は実は BEL + 密度関数の gaussian 近似

BEL average は score の grad だけでなく”non-local” な情報も利用??

- deep calibration への応用. Y が payoff 関数
- Y を path dependent へ拡張 \rightarrow barrier option へのキャリブレーションに必要
- 誤差関数の Lipschitz 性評価 [3]
- neural SDE = infinite dim GAN という解釈が可能 [4]. GAN の正則化テクニックを輸入し, Malliavin 解析で理論計算
- MC estimator を高機能なものに変更した場合の性能比較

Appendix. Adjoint SDE for Calculating SDE

Algorithm 2 Adjoint SDE Method for Calculating \mathcal{S}_s

Require: Simulation path $\{X_t\}$, Brownian increments $\{\delta B_t\}$, where $t \in \{0, \delta t, \dots, T\}$.

- 1: Initialise $\tilde{\mathcal{S}}_T = 0$.
 - 2: **for** $t = T - \delta t$ to 0 (backwards in steps of δt) **do**
 - 3: Calculate \mathcal{D}_t via (36).
 - 4: Update $\tilde{\mathcal{S}}_t = \tilde{\mathcal{S}}_{t+\delta t} + \mathcal{D}_t$.
 - 5: **end for**
 - 6: Compute score process \mathcal{S}_s for $s \in \{0, \delta t, \dots, T\}$ via (37).
-

$$\mathcal{D}_t \approx (\delta t \nabla b(X_t)^\top + \nabla \sigma(X_t)^\top \delta B_t) \tilde{\mathcal{S}}_{t+\delta t} + (\sigma_t^\top(X_t))^{-1} \delta B_t$$

- [1] J. Pidstrigach et al.
Conditioning diffusions using malliavin calculus.
In Proc. ICML, 2025.
- [2] Shinzo Watanabe.
Analysis of wiener functionals (malliavin calculus) and its applications to heat kernels.
The annals of Probability, pp. 1–39, 1987.
- [3] Casey Chu, Kentaro Minami, and Kenji Fukumizu.
Smoothness and stability in gans.
In Proc. ICLR, 2020.
- [4] Patrick Kidger, James Foster, Xuechen Li, Harald Oberhauser, and Terry Lyons.
Neural sdes as infinite-dimensional gans.
In Proc. ICML, 2021.