Nonparametric generative modeling for time series via Schrödinger bridge

Huyên PHAM

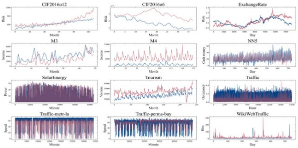
Université Paris Cité (UPC), LPSM

based on joint work with
Mohamed HAMDOUCHE (UPC, Qube RT) and Pierre HENRY-LABORDERE (Qube RT)

Séminaire FIME
Paris, le 8 mars 2024

Generative modeling (for time series)

- Given datasets from an (unknown) **distribution** μ (target) of a time series, e.g.
  - Medical data of a patient
  - Renewable energy production

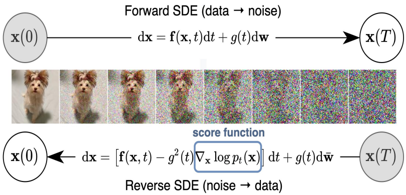


  - Finance: asset/commodity price, ...

- The goal is to
  - generate new (or real-looking) samples of μ:
    - Augmented data added to original training data
    - Useful for improving clinical predictions, weather forecast
    - Financial industry: market stress test, market risk measurement, deep hedging
    - Combined with Reinforcement learning $\rightarrow$ helpful to improve the learning of optimal strategy: Generative Augmented Reinforcement Learning (GARL)

Generative modeling (GM) techniques

- Several competing methods including
  - **Likelihood-based models** (2011-): energy-based models (EBM), variational auto-encoders (VAE), normalizing flow models, etc
  - **Implicit generative models** (2014-): generative adversarial network (GAN)
  - **Score-based diffusion models** (2020-): emergent class of generative AI models that achieved state-of-the-art performance by outperforming GANs.



but mostly for static data/image.

Challenges of GM for time series

A "good" GM for time series data should

- not only learn the time marginals and the joint distribution
- learn the joint distribution while preserving temporal dynamics: respect the causality of variables across time

State-of-the-art generative methods for time series

- **GAN type methods**:
  - Time series GAN (Yoon et al. 19): combination of an *unsupervised adversarial* loss on real/synthetic data and *supervised* loss for generating sequential data
  - Quant GAN (Wiese et al. 20): adversarial generator using temporal convolutional networks
  - Causal optimal transport GAN (Xu et al. 20): adversarial generator using the adapted Wasserstein distance for processes
  - PCF-GAN (Lou et al. 23): Path characteristic function into GAN
  - VolGAN (Vuletic and Cont 23): Arbitrage-free implied volatility surface

- **Neural SDEs**: SDE representation of time series with parametric (e.g. NN) coefficients to be trained for fitting with real samples (Remlinger et al. 21, Kidger et al. 21)

- **Signature** embedding of time series: Fermanian (19), Ni et al. (20), Buehler et al. (20), Morrill et al. (20), etc

▶ Most of these GM are parametric and require the training of NN

State-of-the-art generative methods for time series

- **GAN type methods**:
  - TIME SERIES GAN (Yoon et al. 19): combination of an *unsupervised adversarial* loss on real/synthetic data and *supervised* loss for generating sequential data
  - QUANT GAN (Wiese et al. 20): adversarial generator using temporal convolutional networks
  - CAUSAL OPTIMAL TRANSPORT GAN (Xu et al. 20): adversarial generator using the adapted Wasserstein distance for processes
  - PCF-GAN (Lou et al. 23): Path characteristic function into GAN
  - VOLGAN (Vuletic and Cont 23): Arbitrage-free implied volatility surface

- **Neural SDEs**: SDE representation of time series with parametric (e.g. NN) coefficients to be trained for fitting with real samples (Remlinger et al. 21, Kidger et al. 21)

- **Signature** embedding of time series: Fermanian (19), Ni et al. (20), Buehler et al. (20), Morrill et al. (20), etc

▶ Most of these GM are parametric and require the training of NN

▶ We propose here a nonparametric generative model based on Schrödinger bridge, in the spirit of score-based diffusion models, but over a finite horizon without time reversal, and adapted for time series.

Outline

Reminder on the (classical) Schrödinger bridge (SB) problem

• **Entropy optimal transport** problem of Schrödinger (1932), see survey in Léonard (14): Given:

- reference measure on path spaces (e.g. Wiener $\mathbb{W}$) over a **finite horizon** T
- two distributions μ, ν (e.g. data and prior)

find the closest probability measure $\mathbb{P}$ to the reference w.r.t. Kullback-Leibler divergence, i.e. relative entropy, which admits as marginals: μ at time 0 and ν at time T.

▶ **Stochastic control** formulation by Girsanov's theorem (Dai Pra 1991, Chen et al. 20) Minimize over control process α

$$\mathbb{E}\Big[\frac{1}{2}\int_0^T |\alpha_t|^2 \mathrm{d}t\Big] \qquad (\text{equal to } \mathrm{KL}(\mathbb{P},\mathbb{W}) := \int \log \frac{\mathrm{d}\mathbb{P}}{\mathrm{d}\mathbb{W}} \mathrm{d}\mathbb{P}\ )$$

subject to

$$\mathrm{d}X_t = \alpha_t \mathrm{d}t + \mathrm{d}W_t, \ \ 0 \le t \le T, \qquad X_0 \sim \mu, \quad X_T \sim \nu.$$

Application of SB to generative modeling

The optimal drift of the SB problem is in feedback form: $\alpha_t^* = \mathrm{a}^*(t, X_t)$ with a^* characterized in terms of a Schrödinger system, and the solution can be solved numerically by

- Iterative Proportional Fitting (IPF), a.k.a. Sinkhorn algorithm
- Score-based matching: refinement of IPF

$\rightarrow$ Generative model for sampling μ_{data}: recent works by

De Bortoli et al. (21-), Vargas et al. (21), Wang et al. (21)

Schrödinger bridge time series problem

Let $\mu \in \mathcal{P}((\mathbb{R}^d)^N)$ be the data time series distribution of some $\mathbb{R}^d$-valued process observed at N dates: **target time series measure**.

- **Entropic interpolation of μ:** : Find a diffusion process X on $\mathbb{R}^d$ satisfying

$$\mathrm{d}X_t = \alpha_t \mathrm{d}t + \mathrm{d}W_t, \quad 0 \leq t \leq T, \quad X_0 = 0,$$

with a controlled drift α minimizing

$$J(\alpha) := \mathbb{E}\Big[\frac{1}{2}\int_0^T |\alpha_t|^2 \mathrm{d}t\Big]$$

and such that $(X_{t_1}, \ldots, X_{t_N}) \sim \mu$ (perfect match of the target time series measure), for some time grid $t_0 = 0 < \ldots < t_i < \ldots < t_N = T$.

Remark: the time grid $\mathcal{T} = \{t_i : i \in [\![1, N]\!]\}$ for the interpolation of the Schrödinger diffusion may be different from the observed times of the time series.

Assumptions

Assume that μ admits a density w.r.t. Lebesgue measure on $(\mathbb{R}^d)^N$, denoted by misuse of notation: $\mu(x_1, \ldots, x_N)$.

Denote by $\mu_{\mathcal{T}}^W$ the distribution of Brownian motion W on $\mathcal{T}$, i.e. of $(W_{t_1}, \ldots, W_{t_N})$, hence with density:

$$\mu_{\mathcal{T}}^W(x_1, \ldots, x_N) = \prod_{i=0}^{N-1} \frac{1}{\sqrt{2\pi(t_{i+1} - t_i)}} \exp\Big(- \frac{|x_{i+1} - x_i|^2}{2(t_{i+1} - t_i)} \Big).$$

- We assume that the relative entropy of μ w.r.t. $\mu_{\mathcal{T}}^W$ is finite, i.e.

$$\textbf{(H)} \qquad \mathrm{KL}(\mu|\mu_{\mathcal{T}}^W) := \int \log \frac{\mu}{\mu_{\mathcal{T}}^W} \mathrm{d}\mu < \infty.$$

Remark: Assumption **(H)** is satisfied whenever μ comes from a process with

- Gaussian noise
- Heavy-tailed distribution but with second moment

Solution to Schrödinger bridge time series (SBTS)

Theorem (Diffusion SBTS)

Under **(H)**, the optimal controlled drift of the SBTS problem is in the **path-dependent** form:

$$\alpha_t^* \quad = \quad \mathrm{a}^*(t, X_t; \boldsymbol{X}_{t_1:t_i}), \quad t_i \leq t < t_{i+1}, \quad i = 0, \ldots, N-1,$$

where we set $\boldsymbol{X}_{t_1:t_i} := (X_{t_1}, \ldots, X_{t_i})$, and

$$\mathrm{a}^*(t, x; \boldsymbol{x}_{1:i}) = \nabla_x \log \mathbb{E}_{\mathbb{W}} \Big[\frac{\mu}{\mu_{\mathcal{T}}^W}(X_{t_1}, \ldots, X_{t_N}) \Big| \boldsymbol{X}_{t_1:t_i} = \boldsymbol{x}_{1:i}, X_t = x \Big],$$

for $\boldsymbol{x}_{1:i} = (x_1, \ldots, x_i) \in (\mathbb{R}^d)^i$, $x \in \mathbb{R}^d$. Here $\mathbb{E}_{\mathbb{W}}$ denotes the expectation under which X is a Brownian motion by Girsanov's theorem.

$\rightarrow$ By construction, the diffusion (called SBTS) process

$$\mathrm{d}X_t \quad = \quad \mathrm{a}^*(t, X_t; (X_{t_i})_{t_i \leq t})\mathrm{d}t + \mathrm{d}W_t, \quad X_0 = 0,$$

satisfies $(X_{t_1}, \ldots, X_{t_N}) \sim \mu$.

Application to generative modeling

- Choice of the time grid $\mathcal{T} = \{t_i : i \in [\![1, N]\!]\}$, $\Delta t_i = t_{i+1} - t_i$.
  - When $d = 1$: calibrate Δt_i to the (empirical) variance of μ over $[t_i, t_{i+1}]$ (time-changed Brownian motion):

$$\Delta t_i = \mathrm{Var}_\mu(X_{i+1} - X_i).$$

  - For $d > 1$: normalize each component of the random vector of the time series by its Std, and then use $\Delta t_i = 1$.

- Estimate/learn the Schrödinger drift from samples of μ, see next slides

- Simulate e.g. by Euler scheme the SBTS diffusion $\rightarrow$
  - New samples of μ with realizations of $(X_{t_1}, \ldots, X_{t_N})$
  - Prediction by computing conditional law of $X_{t_{i+1}} | \boldsymbol{X}_{t_1:t_i}$

Alternate expression of the Schrödinger drift function

Using Bayes formula, we derive the following expression:

$$\mathrm{a}^*(t, x; \boldsymbol{x}_{1:i}) = \frac{1}{t_{i+1} - t} \frac{\mathbb{E}_\mu \left[(X_{t_{i+1}} - x) F_i(t, x_i, x, X_{t_{i+1}}) \middle| \boldsymbol{X}_{t_1:t_i} = \boldsymbol{x}_{1:i} \right]}{\mathbb{E}_\mu \left[F_i(t, x_i, x, X_{t_{i+1}}) \middle| \boldsymbol{X}_{t_1:t_i} = \boldsymbol{x}_{1:i} \right]}, \tag{1}$$

for $t \in [t_i, t_{i+1})$, $i = 0, \ldots, N-1$, $\boldsymbol{x}_{1:i} \in (\mathbb{R}^d)^i$, $x \in \mathbb{R}^d$, where

$$F_i(t, x_i, x, x_{i+1}) = \exp\left(-\frac{|x_{i+1} - x|^2}{2(t_{i+1} - t)} + \frac{|x_{i+1} - x_i|^2}{2(t_{i+1} - t_i)} \right).$$

Here $\mathbb{E}_\mu[\cdot|\cdot]$ is the (conditional) expectation under μ $\rightarrow$ One can then estimate the drift function by relying directly on **samples of data distribution** μ.

Remark: When μ is the distribution arising from a Markov chain, then the conditional expectations in (1) (and so the drift function) will depend on the past values $\boldsymbol{X}_{t_1:t_i} = (X_{t_1}, \ldots, X_{t_i})$ only via the last value X_{t_i}.

In practice, we can test the Markov property of μ, and see to what order we need to condition on the past.

Kernel estimation of the drift

• Approximate the conditional expectation under μ by **nonparametric regression** methods, e.g. kernel:

▶ From data samples $\boldsymbol{X}^{(m)}_{1:N} = (X^{(m)}_1, \ldots, X^{(m)}_N)$, $m = 1, \ldots, M$ from μ, the Nadaraya-Watson estimator of the drift function in (1) is given by

$$\hat{\mathrm{a}}(t, x; \boldsymbol{x}_{1:i}) \quad = \quad \frac{1}{t_{i+1} - t} \frac{\sum_{m=1}^{M} (X^{(m)}_{i+1} - x) F_i(t, X^{(m)}_i, x, X^{(m)}_{i+1}) \boldsymbol{K}^i \Big(\frac{\boldsymbol{X}^{(m)}_{1:i} - \boldsymbol{x}_{1:i}}{h} \Big)}{\sum_{m=1}^{M} F_i(t, X^{(m)}_i, x, X^{(m)}_{i+1}) \boldsymbol{K}^i \Big(\frac{\boldsymbol{X}^{(m)}_{1:i} - \boldsymbol{x}_{1:i}}{h} \Big)},$$

for $\boldsymbol{x}_{1:i} = (x_1, \ldots, x_i)$, where $\boldsymbol{K}^i$ is a kernel function on $(\mathbb{R}^d)^i$, e.g. in multiplicative form: $\boldsymbol{K}^i(\boldsymbol{x}_{1:i}) = \prod_{j=1}^{i} K(x_j)$, with K kernel function on $\mathbb{R}^d$, $h > 0$ is the bandwith parameter.

Remarks:

- Choice of kernel is not crucial: we take the quartic kernel $K(x) \propto (1 - |x|^2)^2 1_{|x| \leq 1}$
- Choice of bandwith h is more crucial: tradeoff between bias and variance.
- Plug-in estimate of a^*

SBTS Algorithm

N_π: number of uniform steps in Euler scheme between two observation dates t_i, t_{i+1}:

$$t^\pi_{k,i} \quad = \quad t_i + k/N_\pi, \quad k = 0, \ldots, N_\pi - 1.$$

Algorithm 1: SBTS Simulation

Input: data samples of time series $(X_1^{(\mathrm{m})}, \cdots, X_N^{(\mathrm{m})})$, $m = 1, \ldots, M$, and N_π.
Initialization: initial state $x_0 = 0$;
for $i = 0, \ldots, N-1$ **do**
  Initialize state $y_0 = x_i$;
  for $k = 0, \ldots, N_\pi - 1$ **do**
    Compute $\hat{\mathrm{a}}(t^\pi_{k,i}, y_k; \boldsymbol{x}_{1:i})$ by kernel estimator;
    Sample $\varepsilon_k \in \mathcal{N}(0,1)$ and compute: $y_{k+1} = y_k + \frac{1}{N_\pi}\hat{\mathrm{a}}(t^\pi_{k,i}, y_k; \boldsymbol{x}_{1:i}) + \frac{1}{\sqrt{N_\pi}}\varepsilon_k$;
  end
  Set $x_{i+1} = y_{N_\pi}$.
end
Return: $x_1, \cdots, x_N$ an (approximate) sample of μ

$\rightarrow$ Complexity of order: $O(MNN_\pi)$.

Outline

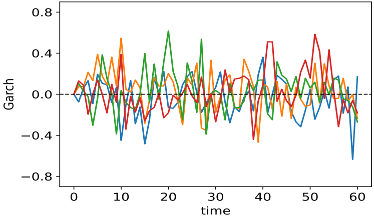
GARCH model

$$\begin{cases} X_{t_{i+1}} & = \sigma_{t_{i+1}} \varepsilon_{t_{i+1}} \\ \sigma^2_{t_{i+1}} & = \alpha_0 + \alpha_1 X^2_{t_i} + \alpha_2 X^2_{t_{i-1}}, \quad i = 1, \dots, N, \end{cases}$$

where $\alpha_0 = 5$, $\alpha_1 = 0.4$, $\alpha_2 = 0.1$, $\varepsilon_{t_i} \sim \mathcal{N}(0, 0.1)$, $N = 60$.

- Parameters: $M = 1000$, bandwith $h = 0.2$, $N_\pi = 100$

Runtime for 1000 generated paths = 100s.



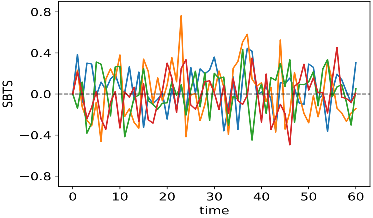


Figure: Samples path of reference GARCH (left) and generator SBTS (right)

Metrics for SBST generator vs GARCH model

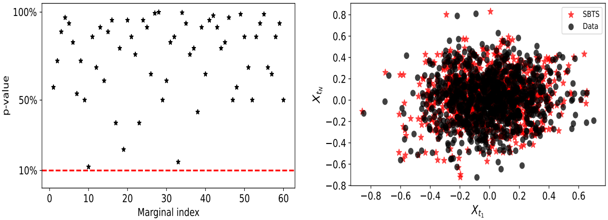


Figure: *Top left*: p-value of the Kolmogorov-Smirnov test for the marginals X_{t_i}. *Top right*: samples plot of the joint distribution (X_{t_1}, X_{t_N}).

A multivariate AR Gaussian model

$$X_{t_{i+1}} = \phi X_{t_i} + \varepsilon_{t_{i+1}}, \quad \text{with} \quad \varepsilon_{t_i} \sim \mathcal{N}(0, \sigma \mathbf{1}_d + (1-\sigma)\mathbb{I}_d),$$

$\phi \in [0,1]$: correlation across time steps, $\sigma \in [-1,1]$: correlation across the components.

▶ We compute the predictive score: Mean absolute error between conditional mean (from generated model) and the true value: $\mathbb{E}[X_{t_{i+1}} | \boldsymbol{X}_{t_i}] = \phi X_{t_i}$.

	Temporal correlation (fixing $\sigma = 0.8$)			Feature correlation (fixing $\phi = 0.8$)		
Settings	$\phi = 0.2$	$\phi = 0.5$	$\phi = 0.8$	$\sigma = 0.2$	$\sigma = 0.5$	$\sigma = 0.8$
Predictive score (lower the better)						
SBTS	**.161 ± .016**	**.180 ± .026**	**.244 ± .014**	.325 ± .052	.295 ± .038	**.244 ± .014**
TimeGAN	.640 ± 0.003	0.412 ± 0.002	.251 ± .002	**.282 ± .005**	**.261 ± .002**	.251 ± .002

Table: Predictive score for SBTS vs TimeGan

Fractional Brownian motion

Fractional Brownian motion (FBM) with Hurst index $H = 0.1$.

• Parameters: $M = 1000$, $N = 60$, $N_\pi = 100$, bandwith $h = 0.05$.
Runtime for 1000 generated paths = 100s.

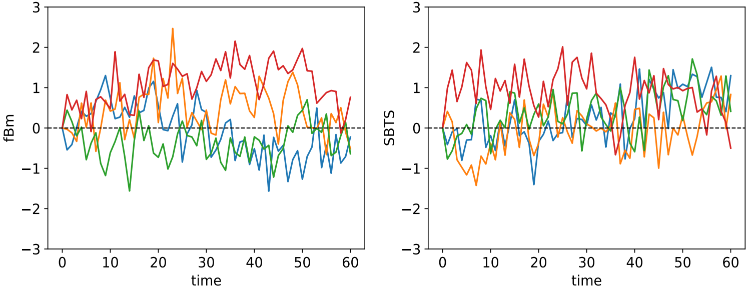


Figure: Four samples path of reference FBM (left) and generator SBTS (right)

Metrics for SBST generator vs FBM

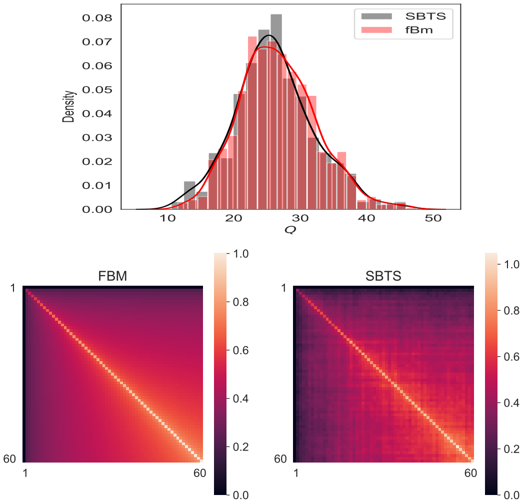


Figure: *Top*: Quadratic variation distribution $\sum_{i=1}^{N} |X_{t_{i+1}} - X_{t_i}|^2$ for $N = 60$.
Bottom: Covariance matrix for reference FBM and SBTS

Estimation of Hurst index

Standard estimator of Hurst index:

$$\hat{H} = \frac{1}{2}\left[1 - \frac{\log\left(\sum_{i=0}^{N-1} |X_{t_{i+1}} - X_{t_i}|^2\right)}{\log N}\right].$$

- From our generated SBTS with $N = 60$, we get:

$$\hat{H} = 0.102, \quad \text{Std} = 0.003.$$

Application to deep hedging on real data sets (SPX)

Data: Index prices of SPX from jan. 1, 2010 to jan. 30, 2020, with sliding window of $N = 5$ days, $\rightarrow M = 2500$ samples.

• Consider a ATM call option on SPX: $g(X_T) = (X_T - K)_+$, and we search for a price p^* and hedging strategy Δ^* minimizing the quadratic criterion (loss function):

$$(p, \Delta) \quad \mapsto \quad \mathbb{E}\Big| \underbrace{p + \sum_{i=0}^{N-1} \Delta_{t_i}(X_{t_{i+1}} - X_{t_i}) - g(X_T)}_{\text{PnL}} \Big|^2 = \text{replication error}$$

Application to deep hedging on real data sets (SPX)

Data: Index prices of SPX from jan. 1, 2010 to jan. 30, 2020, with sliding window of $N = 5$ days, $\to M = 2500$ samples.

• Consider a ATM call option on SPX: $g(X_T) = (X_T - K)_+$, and we search for a price p^* and hedging strategy Δ^* minimizing the quadratic criterion (loss function):

$$(p, \Delta) \quad \mapsto \quad \mathbb{E}\Big| \underbrace{p + \sum_{i=0}^{N-1} \Delta_{t_i}(X_{t_{i+1}} - X_{t_i}) - g(X_T)}_{\text{PnL}} \Big|^2 \; = \; \text{replication error}$$

▶ We parametrize Δ by a LSTM network that is trained from synthetic data sets produced by SBTS (10 times more), and we compare the results with real-data sets.

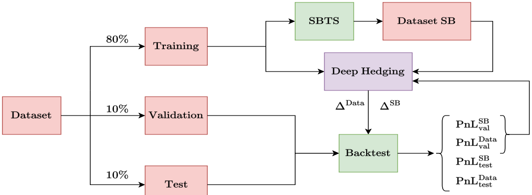


Figure: Procedure of backtest for deep hedging

Comparison of the PnL and replication error with real-data and generative SBTS

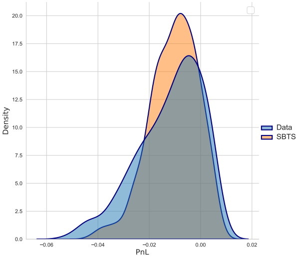


Figure: Deep hedging PnL distribution from test set

	Premium	Mean PnL	Std PnL
Data	0.0059	-0.0119	0.0124
SBTS	0.0078	-0.0101	0.0114

Table: Price, Mean of PnL and its Std (replication error) on the test set.

Concluding remarks

- Novel generative model for time series based on Schrödinger bridge (SB) approach:
  - Solution described by a forward stochastic differential equation (SDE) over a finite period, which matchs perfectly the data distribution: bridge between data-driven model and classical diffusion model-based approach.
  - Drift estimated by nonparametric regression, e.g. kernel method: practical and low-cost computationally (plug-in estimate that does not require training of neural networks as in GAN type methods)

- Series of numerical experiments, including financial applications with real-data, to illustrate the performance and accuracy of our generative SBTS.

- Further developments:
  - SBTS model can be enriched to fit more accurately with data time series:
    - diffusion coefficient
    - jump-diffusion process
  - Diffusion SB valued in functional space in view of applications to implied volatility surface generation
  - Kernel method suffer from curse of dimensionality. Alternately, the drift function can be approximated by neural networks, and more precisely with a LSTM architecture.

Reference

M. Hamdouche, P. Henry-Labordère, H. Pham. Generative modeling for time series via Schrödinger bridge. SSRN 4412434, arXiv:2304.05093

Code available on Github: https://github.com/hamdouchm/SBTimeSeries