Conditional Sig-Wasserstein Generative models

to generate realistic synthetic time series

Hao Ni

UCL, the Turing Institute and DataSig

DataSig

A rough path between
mathematics and data science

University of Oxford · The Alan Turing Institute · Imperial College London · UCL · UKRI Engineering and Physical Sciences Research Council

Outline

Time series generation

- Paper: Conditional Sig-Wasserstein GANs for Time Series Generation.
- Joint work with Lukasz Szpruch, Shujian Liao, Magnus Wiese and Baoren Xiao.
- Code are available at GitHub: `https://github.com/SigCGANs/Conditional-Sig-Wasserstein-GANs.git`

Objectives

- to build a high-quality *conditional generative model* for time series generation to better capture the heterogeneity of time series $X_{1:T}$.
- to improve the performance and training stability of the Wasserstein Generative Adversarial Networks using the *signature of the path*.

Conditional Generative Model for Time Series

To model the joint distribution of $X_{[1,T]}$ effectively, we aim to learn the conditional distribution $\mathbb{P}(X_{t,\text{future}}|X_{t,\text{past}})$ from data.

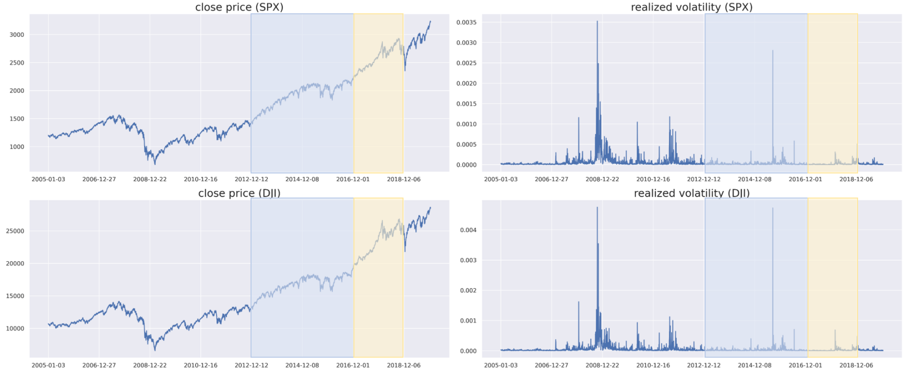


Figure: An example of 4-dimensional financial time series composed of the price and realized volatlity of SPX and DJI from 2005-01-01 to 2018-12-31. The blue region represents the past time series and the yellow region represents the future time series.

Wasserstein Generative Adversarial Networks

Wasserstein-1 metric (W_1)

Let $\mu, \nu \in \text{Prob}(\mathcal{X})$ with a compact support K. The Kantorovich and Rubinstein dual representation of Wasserstein-1 metric is given by

$$W_1(\mu,\nu) = \sup_{\text{continuous } f:\mathcal{X}\to\mathbb{R}, Lip(f)\leq 1} \mathbb{E}_{\tilde{X}\sim\mu}[f(\tilde{X})] - \mathbb{E}_{\tilde{X}\sim\nu}[f(\tilde{X})].$$

Wasserstein Generative Adversarial Networks (WGAN)

- Given samples $(X^{(i)})_{i=1}^{N}$ sampled from the true distribution $p^*(X)$.
- Latent variable Z: a $\mathcal{Z}$-valued random variable.
- Goal: To train a model such that for $g_\theta : \mathcal{Z} \times \Theta \to \mathcal{X}$ so as to

$$\min_{\theta} \boxed{\max_{\alpha} \mathbb{E}[f_\alpha(g_\theta(Z))] - \mathbb{E}_{X\sim p^*}[f_\alpha(X)]}.$$

Outline

The signature of a path

Definition (Signature of a path)

Let X denote a $\mathbb{R}^d$-valued path of bounded 1-variation. The signature of the path X is defined as $S(X_J) = (1, X_J^1, \cdots, X_J^k, \cdots) \in T((\mathbb{R}^d))$, where

$$X_J^k = \int_{t_1 < t_2 < \cdots < t_k, t_1, \cdots, t_k \in J} dX_{t_1} \otimes \cdots \otimes dX_{t_k}. \quad (1)$$

$S_M(X_J) = (1, X_J^1, \cdots, X_J^M))$ is the truncated signature up to degree M.

Embedding time series to the path space

- There are different ways to embed time series to the path space.
- In our work, we choose to embed discrete time series X to a time jointed path as defined in [1] as this embedding ensures the uniqueness of the signature.

The signature of a path

The signature of a path

- is a graded infinite series to summarize the path (time series) faithfully.
- is a *universal* basis for continuous functions on the path space.[2]

ESig, Signatory[3] and iisignature[4] are three Python libraries for signature computation.

The Signature Wasserstein-1 metric (Sig-W_1)

We propose to define the truncated Signature Wasserstein-1 metric (Sig-W_1) up to degree M as follows:

$$\text{Sig-}W_1^{(M)}(\mu, \nu) = |\mathbb{E}_\mu[S_M(X)] - \mathbb{E}_\nu[S_M(X)]|, \tag{2}$$

where μ and ν are two measures on the path space and $|.|$ is l_2 norm.

Main idea

$$W_1(\mu, \nu) = \sup_{\text{continuous } f:\mathcal{X}\to\mathbb{R}, Lip(f)\leq 1} \mathbb{E}_{\tilde{X}\sim\mu}[\underbrace{f(\tilde{X})}_{\approx L(S(\tilde{X}))}] - \mathbb{E}_{\tilde{X}\sim\nu}[\underbrace{f(\tilde{X})}_{\approx L(S(\tilde{X}))}].$$

Remark

In [5], if one chooses the truncated signature up to degree M as the feature map, then the corresponding Maximum Mean Discrepancy (Sig-MMD) is the square of Sig-$W_1^{(M)}(\mu, \nu)$.

Outline

1. Motivation and Objective
2. Sig-W1 metric
3. The Signature-based Conditional WGAN (SigCWGAN)
4. Numerical Results

The Signature-based Conditional WGAN (SigCWGAN)

Problem setup

- We assume that a $\mathbb{R}^d$-valued time series $(X_t)_{t=1}^T$ satisfies

$$X_{t+1} = f(X_{t-p+1:t}) + \varepsilon_t,$$

where $\mathbb{E}[\varepsilon_{t+1}|\mathcal{F}_t] = 0$, $\mathcal{F}_t$ is the information up to time t and $f : \mathbb{R}^{p \times d} \to \mathbb{R}^d$ is a continuous but unknown function.
- The objective of SigCWGAN is to generate synthetic time series whose condition distribution is as close as to the joint distribution of $x_{\text{future}} = X_{t+1:t+q}$ given the past time series $x_{\text{past}} = X_{t-p+1:t}$.

The Conditional AR-FNN Generator

Given $X_{t-p+1:t}$, estimate the next step $\hat{X}^{(t)}_{t+1} = g_\theta(X_{t-p+1:t}, Z_{t+1})$. Then use $\hat{X}^{(t)}_{t+1}$ to generate the step-2 estimator by $\hat{X}^{(t)}_{t+2} = g_\theta(X_{t-p+2:t}, \hat{X}^{(t)}_{t+1}, Z_{t+2})$ and repeating this procedure until obtaining the step-q estimator $\hat{X}^{(t)}_{t+1:t+q}$.

Conditional Sig-W_1 Discriminator

We define the truncated conditional Signature Wasserstein-1 metric of degree M denoted by C-Sig-$W_1^{(M)}$ on μ and ν as follow:

$$\text{C-Sig-}W_1^{(M)}(\mu,\nu|x_{\text{past}}=x) := |\mathbb{E}_\mu[S_M(X_{\text{future}})|x_{\text{past}}=x] - \mathbb{E}_\nu[S_M(X_{\text{future}})|x_{\text{past}}=x]|.$$

Loss function

$$L(\theta) = \sum_t |\mathbb{E}_\mu[S_M(X_{t+1:t+q})|X_{t-p+1:t}] - \mathbb{E}_\nu[S_M(\hat{X}^{(t)}_{t+1:t+q})|X_{t-p+1:t}]|,$$

where ν and μ denote the conditional distribution induced by the real data and synthetic generator respectively, g_θ is the generator, $\hat{X}^{(t)}_{t+1:t+q}$ is the q-step forecast generated by g_θ.

SigCWGAN Algorithm

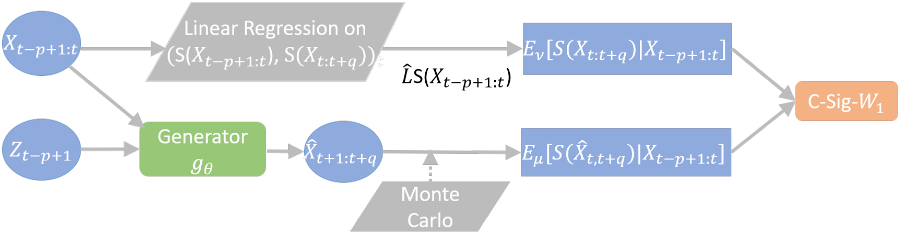


Figure: The illustration of the flowchart of SigCWGAN.

Conditional WGAN vs Conditional SigWGAN

Challenges of the conditional WGAN

$$\text{C-W}_1(\mu, \nu | X_{\text{past}}) = \max_{|f_\alpha|_{\text{Lip}} \leq 1} \boxed{\mathbb{E}_\mu[f_\alpha(X_{\text{future}})|X_{\text{past}}]} - \mathbb{E}_\nu[f_\alpha(X_{\text{future}})|X_{\text{past}}].$$

The estimator for $\mathbb{E}_\mu\left[f_\alpha(X_{\text{future}})|X_{\text{past}} = x_{t-p:t}\right]$ has two choices:

- $f_\alpha(x_{t+1:t+q})$ (noisy estimator). Under the true measure μ, given $X_{past} = x_{t-p:t}$, it is very likely that there is only one corresponding sample of the future path.
- Regress $(x_{t-p:t}, f_\alpha(x_{t+1:t+q}))_{p+1}^{T-q}$ to obtain the estimator for the conditional expectation (heavy computation).

C-Sig-WGAN

- By embedding time series to the *signature* space, the conditional W_1 metric can be approximated by

$$\text{C-Sig-}W_1^{(M)}(\mu, \nu) = |\mathbb{E}_\mu[S_M(X)|X_{\text{past}}] - \mathbb{E}_\nu[S_M(X)|X_{\text{past}}]|, \tag{3}$$

  where μ and ν are two measures on the path space and $|.|$ is l_2 norm.
  No optimisation needed.
- We add the supervised learning module to learn $\mathbb{E}_\mu[S_M(X)|X_{past}]$, which is one-off and can be done prior to the GAN learning as S_M is the deterministic mapping.

Outline

Experiment Setup

Datasets

- Synthetic Data: Vector Autoregressive model;
- Empirical Data: multi-dimensional time series of both the log return of the close prices and the log of median realised volatility of (a) the SPX only; (b) the SPX and DJI. [a]

[a] It is retrieved from the Oxford-Man Institute's "realised library"[6].

Baselines

To benchmark with SigCWGAN, we choose three representative generative models for the time-series generation, i.e.

1. TimeGAN [7];
2. RCGAN [8];
3. GMMN [9].

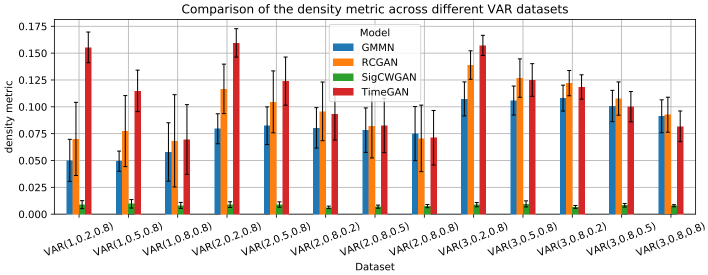
Test Metrics

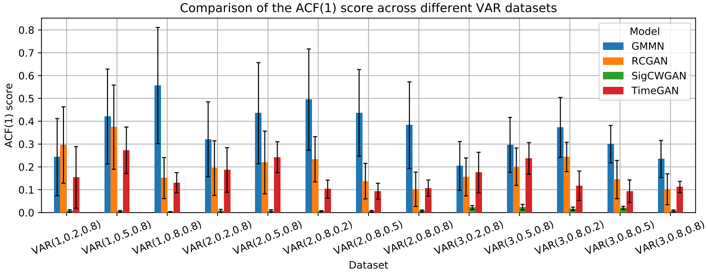
We consider three main criteria:

(a) the marginal distribution of time series;

(b) the temporal and feature dependence;

(c) usefulness[7] - synthetic data should be as useful as the real data when used for the same predictive purposes (i.e. train-on-synthetic, test-on-real(TSTR), train-on-real, test-on-real(TRTR)).


Comparison of the density metric across different VAR datasets
Model
GMMN
RCGAN
SigCWGAN
TimeGAN
density metric
0.000
0.025
0.050
0.075
0.100
0.125
0.150
0.175
VAR(1,0.2,0.8)
VAR(1,0.5,0.8)
VAR(1,0.8,0.8)
VAR(2,0.2,0.8)
VAR(2,0.5,0.8)
VAR(2,0.8,0.2)
VAR(2,0.8,0.5)
VAR(2,0.8,0.8)
VAR(3,0.2,0.8)
VAR(3,0.5,0.8)
VAR(3,0.8,0.2)
VAR(3,0.8,0.5)
VAR(3,0.8,0.8)
Dataset


Comparison of the ACF(1) score across different VAR datasets
Model
GMMN
RCGAN
SigCWGAN
TimeGAN
ACF(1) score
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
VAR(1,0.2,0.8)
VAR(1,0.5,0.8)
VAR(1,0.8,0.8)
VAR(2,0.2,0.8)
VAR(2,0.5,0.8)
VAR(2,0.8,0.2)
VAR(2,0.8,0.5)
VAR(2,0.8,0.8)
VAR(3,0.2,0.8)
VAR(3,0.5,0.8)
VAR(3,0.8,0.2)
VAR(3,0.8,0.5)
VAR(3,0.8,0.8)
Dataset

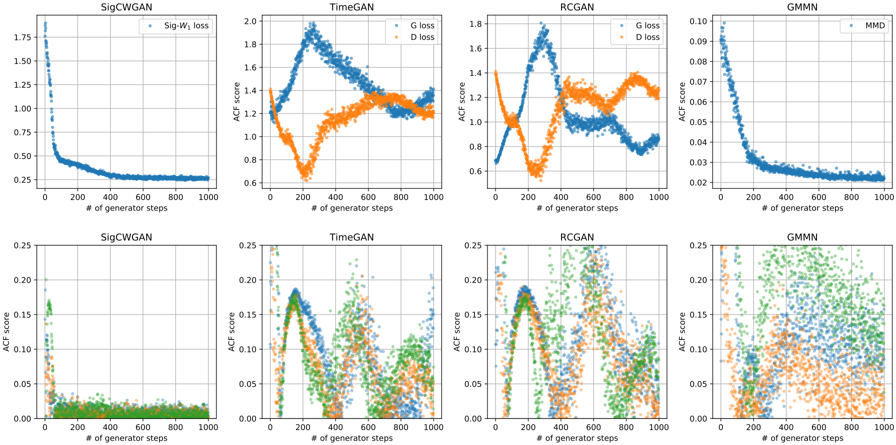


Figure: (Upper panel) Evolution of the training loss functions. (Lower panel) Evolution of the ACF scores. Each colour represents the ACF score of one dimension. Results are for the 3-dimensional $\mathrm{VAR}(1)$ model for $\phi = 0.8$ and $\sigma = 0.8$.

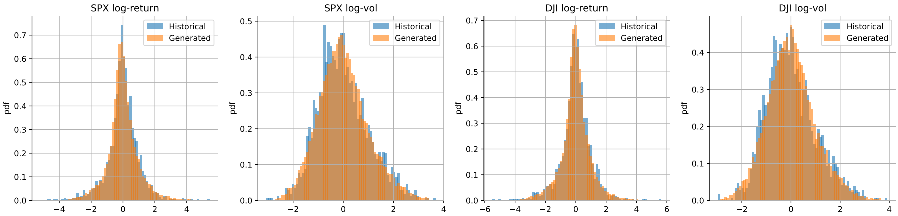


Figure: Comparison of the marginal distributions of the generated SigCWGAN paths and the SPX and DJI data.

Metrics	marginal distribution	auto-correlation	correlation	$R^2(\%)$	Sig-W_1
SigCWGAN	0.01730, 0.01674	0.01342, **0.01192**	**0.01079**, **0.07435**	2.996, 7.948	**0.18448**, **4.36744**
TimeGAN	0.02155, 0.02127	0.05792, 0.03035	0.12363, 0.61488	5.955, 8.586	0.58541, 5.99482
RCGAN	0.02094, **0.01655**	0.03362, 0.04075	0.04606, 0.15353	**2.788**, **7.190**	0.47107, 5.43254
GMMN	**0.01608**, 0.02387	**0,01283**, 0.02676	0.04651, 0.22380	9.049, 7.384	0.59073, 6.23777

Table: Numerical results of the stock datasets. In each cell, the left/right number are the result for the SPX data/ the SPX and DJI data respectively. We use the relative error of TSTR R^2 against TRTR R^2 as the R^2 metric.

Daniel Levin, Terry Lyons, and Hao Ni.
Learning from the past, predicting the statistics for the future, learning an evolving system.
arXiv preprint arXiv:1309.0260, 2013.

Terry Lyons.
Rough paths, signatures and the modelling of functions on streams.
arXiv preprint arXiv:1405.4537, 2014.

Patrick Kidger and Terry Lyons.
Signatory: differentiable computations of the signature and logsignature transforms, on both cpu and gpu.
arXiv preprint arXiv:2001.00706, 2020.

Jeremy Reizenstein and Benjamin Graham.
The iisignature library: efficient calculation of iterated-integral signatures and log signatures.
arXiv preprint arXiv:1802.08252, 2018.

Ilya Chevyrev and Harald Oberhauser.

Signature moments to characterize laws of stochastic processes.
arXiv preprint arXiv:1810.10971, 2018.

Gerd Heber, Asger Lunde, Neil Shephard, and Kevin Sheppard.
Oxford-man institute's realized library, version 0.3, 2009.

Jinsung Yoon, Daniel Jarrett, and Mihaela van der Schaar.
Time-series generative adversarial networks.
In *Advances in Neural Information Processing Systems*, pages 5509–5519, 2019.

Stephanie Hyland, Cristóbal Esteban, and Gunnar Rätsch.
Real-valued (medical) time series generation with recurrent conditional gans.
2018.

Yujia Li, Kevin Swersky, and Rich Zemel.
Generative moment matching networks.
In *International Conference on Machine Learning*, pages 1718–1727, 2015.