Scenario Generation and Data Augmentation in QWIM

Mingyue Zhang Anna Zhang Siyuan Zou

Adivsor: Cristian Homescu

1 Introduction

In finance world, there is always a need to conduct scenario analysis on the portfolio, so that people can make as wise decisions as possible. Specifically, "tail events" are the ones that we care about most: we need to know and prepare for the worst cases. Statistical methods such as VaR have been proved inefficient in tackling this kind of problem, so professionals and scholars turn to machine learning approaches and try to construct plausible distributions and scenarios using them. With the augmented data in hand, not only can we use it for the purpose of risk management, but also we can conduct portfolio management, test investment strategies, tackle unbalanced dataset and so on. Therefore, scenario generation and data augmentation are indeed valuable topics that we should pay attention to.

Among all kinds of machine-learning based methods, Restricted Boltzmann Machine (RBM), Variational Autoencoder (VAE) and generative adversarial network (GAN) seem to be promising and can produce some good results.

RBM is a two-layer neural network which has a restricted constraint on the structure of network: connections between any two units within the same layer are forbidden. By optimizing on the network's weights and biases, this model tries to reconstruct the inputs of the training dataset so that it can learn the joint distribution and hence produce reasonable augmented data. Since the result from parametric model is poor, Bernoulli RBM is used in [1] and is trained on the daily log-return data of FX rates ranging from 1999 to 2019. Moreover, the Bernoulli RBM captures the dependency structure of the time series: it can generate similar tail dependence functions which are close to the real ones. The flexibility of the model allows us to replicate the non-stationarity within the data.

Generative adversarial network (GAN) is another approach that has achieved lots of appealing results on scenario generations and data augmentation. The basic idea [2] of this method to use the discriminator to judge whether the inputs are collected from the real distribution or are produced by the generator. If a generator can successfully "deceive" the discriminator, then we could use it to augment the real-world data. The study of Takahashi et al. [3] is the first proposal to use this kind of model on financial time series. FIN-GANs are developed to generate data which remain the major stylized facts

of the real data such as the linear unpredictability, the fat-tailed price return distribution, volatility clustering, the leverage effects, the coarse-fine volatility correlation, and the gain/loss asymmetry. Multivariate financial time series generation is handled in [4]. The cross-correlation matrices for log-implied volatilities and implied volatility log-returns of generated and historical data are similar to each other. In [5], Wiese et al. propose Quant GAN model, which uses temporal convolutional networks (TCNs) as the generator. The TCN generator enables the model to construct long-term dependencies in sequences and guarantees stationarity. It also avoids the problem of exponentially vanishing or exploding gradients within RNNs. The authors successfully trained the model on the SP data and showed that QuantGAN can precisely model the distributional and dependence properties of the real data. To capture the temporal dependence of joint probability distributions induced by time-series data, the signature is used in [6]. The signature of a path can be derived in lots of dimensions and provides us a comprehensive description of the data: lower order terms give us the global description and higher order term shed some light on the local structure of the path. The conditional Sig-Wasserstein GAN for time series generation based on the explicit approximation of W1 metric using the signature features space is developed in the article and this approach simplifies the training process.

Facing the scarce data environment, Variational Autoencoder (VAE) based model is developed in [7]. VAEs are stable and flexible generators for scarce data environments and can be work consistently well under different differential operators and architectures. What's more, log-signature takes place of the signature so that a linear space that describes the characteristics of price path can be created. The features of SP data can be captured by this method and the training process is efficient.

2 Methodology

2.1 Background Theory: The Signature Method for Time Series

The signature of a path, which is a key concept from the rough path theory, characterises the law of a stochastic process. We give the definition in [6] below.

Definition 2.1 (Signature) *Let $X \in C_0^p(J, \mathbb{R}^d)$ such that the following integration makes sense. The signature of the path X is defined as $S(X_J) = (1, X_J^1, ..., X_J^k, ...) \in T((\mathbb{R}^d))$, where*

$$X_J^k = \int_{t_1 < t_2 < ... < t_k, t_k \in J} dX_{t_1} \otimes \dots dX_{t_k} \tag{1}$$

Let $X_M(S_J)$ denote the truncated signature of X of degree M, i.e., $S_M(X_J) = (1, X_J^1, ..., X_J^M)$.

$T((\mathbb{R}^d)) := (a_0, a_1, a_2, \dots) : a_n \in (\mathbb{R}^d)^{\otimes n} \forall n \geq 0$ denotes the tensor algebra space where the signature of a $\mathbb{R}^d$-valued path takes value.

The signature extraction approach has several benefits. First, the mapping from the path space X to the signature space uniquely determines X up to time parameterization (the uniqueness property) [8]. Second, non-linear continuous functions of the path can be universally approximated by linear functions in the signature space (the universality property) [9]. Third, a truncated signature up to a finite degree can approximate the

signature of a path (which is infinite) reasonably well due to its factorial decay property [6]. Concretely, lower order terms of the signature gives a global description of the path whereas higher order terms captures the locality information.

2.2 Data Transformation

Lead-lag transform To apply the signature concept to feature extraction of time series, the first step is to embed the (discrete) data streams into a (continuous) path. We apply the lead-lag transform and the resulting path is a piece-wise linear continuous one, which makes the calculation of the signature computationally fast [9]. The lead-lag transformation of the data stream is defined as

$$\hat{S}_t := \begin{cases} (S_{t_i}, S_{t_{i+1}}) & \text{for } t \in [2i, 2i+1) \\ (S_{t_i}, S_{t_{i+1}} + 2(t-(2i+1))(S_{t_{i+2}} - S_{t_{i+1}})) & \text{for } t \in [2i+1, 2i+\frac{3}{2}) \\ (S_{t_i} + 2(t-(2i+\frac{3}{2}))(S_{t_{i+1}} - S_{t_i}), S_{t_{i+2}}) & \text{for } t \in [2i+\frac{3}{2}, 2i+2) \end{cases}$$

for $t \in [0, 2N]$ [10].

Signature and log-signature transforms It is explained in the paper [7] that generative modelling on signature space may be problematic since the signature space is not linear. The implication is that small perturbations (as one might expect from the output of a generator) of the signature of a path will in general not correspond to the signature of some other path. They work around with this issue by targeting the log-signature instead, which essentially captures the same information as the signatures but now spans a linear space. Moreover, the log-signature is a more sparse representation of the path which is analogous to dimensionality reduction in a machine learning context. We work with both signatures and log-signatures to compare such impact on the output quality.

2.3 Model Training: Signature-based Conditional GAN vs. VAE

As a starting point, we explore univariate time series forecasting using generative modeling. Specifically, we use the QuantGAN proposed by [5] and analyze results in the next section. Details of the QuantGAN architecture can be found in the authors' original paper. We proceed to discuss the multi-dimensional time series forecast using the signature approach.

The objective of the model is to generate the joint distribution of the future time series given the past. First, we experiment with a Sig-CWGAN proposed by [6].

The architecture of the Generative Adversarial Network (GAN) consists of a (conditional) generator $G^\theta : \mathbb{R}^{d\times p} \times Z \to \mathbb{R}^d$ and a discriminator D. The generator takes in the past data with a window size of p and a random noise vector (standard normal) Z to generate the next step forecast $\hat{X}_{t+1}^{(t)}$, then $\hat{X}_{t+1}^{(t)}$ is used to generate the step-2 forecast, etc. The q-step forecast is obtained by repeating the procedure q times. To account for non-stationarity in our data, the Auto-regressive (AR) model from classical time series analysis is chosen as the generating function, combined with a simple feed-forward neural network (AR-FNN). By doing so, the proposed conditional AR-FNN generator is able to capture the autoregressive structure of the time series by construction, and with a lagged $p = 3$ values as the conditioning variable, the generator is able to achieve great resemblance of temporal dependence to real data.

Implied by its name, the discriminator's task is to discriminate between real and fake data. Here we use a conditional Sig-W_1 architecture, where Sig-W_1 is a distance measure proposed in [6]:

$$\text{C-Sig-}W_1^{(M)} := \left|\mathbb{E}_\mu \left[S_M(X_{future}) \left| x_{past} = x\right.\right] - \mathbb{E}_\nu \left[S_M(X_{future}) \right| x_{past} = x\left.\right]\right| \tag{2}$$

The motivation derives from a theorem [11] which states that if the expected signature of two random variables X and Y are the same (with infinite radius of convergence), then $X = Y$ in the distribution sense. Combined with the universality of the signature map, the authors in [6] arrived at a new metric that works with the signature feature space rather than the path space:

$$\text{Sig-}W_1(\mu, \nu) := \sup_{|L|<1, \text{L is a linear functional}} L(\mathbb{E}_\mu[S(X)] - \mathbb{E}_\nu[S(X)])$$

The advantage is that it reduces the nonlinear optimisation of computing W_1 distance over the class of Lipschitz functionals to the linear one over the linear functionals on the signature space. Furthermore, when the norm of L is chosen as the l_2 norm of the linear coefficients of L, the above optimization has an analytic solution (see details in [6]).

Given two conditional distributions $\mu(X_{future}|x_{past})$ and $\nu(X_{future}|x_{past})$, the discriminator aims to detect whether they are the same by quantifying the distance (l_2 norm) between them. The loss function is just the summation of equation (2) over each time t, i.e.

$$L(\theta) = \sum_t \left|\mathbb{E}_\nu \left[S_M(X_{t+1:t+q}) \left| X_{t-p+1:t}\right.\right] - \mathbb{E}_\mu \left[S_M(\hat{X}^{(t)}_{t+1:t+q}) \right| X_{t-p+1:t}\left.\right]\right| \tag{3}$$

where μ and ν denote the conditional distribution induced by the generator and real data, respectively.

Next, we experiment with using Variational Autoencoder as the generative model, a Sig-CVAE proposed by [7]. VAE consists of an encoder network and a decoder network. The encoder network in our experiment has one hidden layer, two latent layers (50 nodes on each), a leaky parametric ReLU activation function with paramater $\alpha = 0.3$. The decoder has one hidden layer (50 nodes) and the same activation function as the encoder. To further account for nonstationarity in time series, the model is refined to incorporate the previous market conditions. To evaluate the to how well the generated distribution match the real distribution, a computationally efficient Maximum Mean Discrepancy (MMD) metric for laws of stochastic processes proposed by [12] is chosen. The authors use this metric to develop a two-sample test for stochastic processes, which can evaluate the quality of the generator. More specifically, one can compute the signature-based MMD test statistic

$$T(X_1, \ldots, X_n; Y_1, \ldots, Y_n) := \frac{1}{n(n-1)} \sum_{i,j;i\neq j} k(X_i, Y_j) + \frac{1}{n(n-1)} \sum_{i,j;i\neq j} k(Y_i, Y_j)$$

where $k(\cdot, \cdot)$ denotes the signature kernel ([12], Proposition 4.2). Then one can obtain the threshold $c_\alpha := 4\sqrt{-n^{-1} \log \alpha}$ (see details in [7]).

The motivation for using a VAE is that it is a more stable and flexible generator for small data environment compared to GAN, which is very difficult to train and requires huge

amount of data for learning. Further, VAE may attribute a high probability to (nearby) points other than the ones in the training set [7], as it tends to have inferior image processing ability compared to GAN. However, this is not a concern for our purpose in generating different market scenarios, as ideally one would like the model to generate both accurate and meaningful unseen market scenarios.

2.4 Postprocessing of the outputs of VAE

Since the output of the generator are log-signatures, one might want to convert them back into the path space for purposes such as portfolio management. In order to invert (log-)signatures into paths, one possible method to do so is using the evolutionary algorithm [7, 13]. However, we note that this step in our project prove to be very computationally expensive, and the task of developing a computationally efficient way for signature inversion is subject to ongoing research.

3 Performance Evaluation

We use several evaluation metrics proposed in [7, 5] to measure both the distributional and dependence properties of the generated data. We describe succinctly the motivation and formula for each below (for a more detailed description, see [7, 5]) and discuss our results in the next section.

3.1 Distributional Scores

Wasserstein-1 distance To evaluate the two generative models, we first choose a distributional metric which looks at the difference between the mean function values of two samples. The W_1 distance is also called the Earth Mover Distance, for loosely speaking it describes how much probability mass has to be moved to transform the generated distribution into the real distribution.

3.2 Dependence Scores

ACF score The ACF score compares the autocorrelation of the historical vs the generated time series. Denote the autocorrelation function by $C(\tau; r) = \mathrm{Corr}(r_{t+\tau}, r_t)$, the ACF score is computed for a function $f : \mathbb{R} \to \mathbb{R}$ as

$$\mathrm{ACF}(f) := \|C(f(r_{1:T})) - \frac{1}{M}\sum_{i=1}^{M} C(f(r^{(i)}_{1:T,\theta}))\|_2$$

Leverage effect The leverage effect is measured using the correlation of the lagged squared log returns and the log returns themselves, i.e. $L(\tau; r) = \mathrm{Corr}(r^2_{t+\tau}, r_t)$. and the levereage effect score is defined by

$$\|L(r_{1:T}) - \frac{1}{M}\sum_{i=1}^{M} C(r^{(i)}_{1:T,\theta})\|_2$$

4 Empirical Results

4.1 The Data

The datasets were selected to have the following features: be good proxies for most representative asset and sub-asset classes, widely available, high liquidity, daily granularity, encompass as many market regimes as possible, etc..

To meet the above criteria, we choose the following assets/sector Indexes:

- S&P 500 Energy Index (S5ENRS)
- S&P 500 Information Technology Index (S5INFT)
- S&P 500 Financials Sector GICS Level 1 Index (S5FINL)
- S&P 500 Consumer Staples Index (S5CONS)
- Bloomberg Barclays US Aggregate Bond Index (LBUSTRUU)

We chose indexes since they tend to have "nicer" statistical properties compared to time series of individual stocks or bonds. The available data are daily historical price ranging from 1990/01/01 – 2009/05/01 (exported from Bloomberg)

4.2 Data Preprocessing

In the signature-based Conditional GANs (VAE) model, the full time series are subdivided into equal length intervals. We use 20 days as segment which is corresponding to a month. After calculating monthly log returns, we convert the samples into log signature. Conditioning on one month's data, the future returns paths can be simulated.

4.3 Results

Figure 1 shows the results of our first stage model QuantGANs, which is temporal convolutional networks (TCNs). We use the model to simulate log returns of the data and compare it with the original one. The orange shade indicates the difference between the historical data and our generated data.

Figure 2 shows the comparison for the unconditional signature path method. The five rows (d=5) are corresponding to five indices we choose. We can see that the generated paths have very close shape of the original ones, and the tail events can be captured. Also, ACF score and Sig-W1 distance converge towards zero, which proves the advantages of using signature-based GANS and Sig-W1 distance as loss function.

To illustrate that the accuracy of using conditional VAE model, we plot the projections of generated monthly log-signature paths (Figure 3). We transfer the high-dimensional signatures into several 2-D graphs.

To better visualize the results of our generated signature paths, we choose some of the generated paths to compare with the historical (real) time series. In figure 4, we can see that some of the synthetic time series have similar tendency as the real path. And in figure 5, the generated paths have close distribution of the real paths.

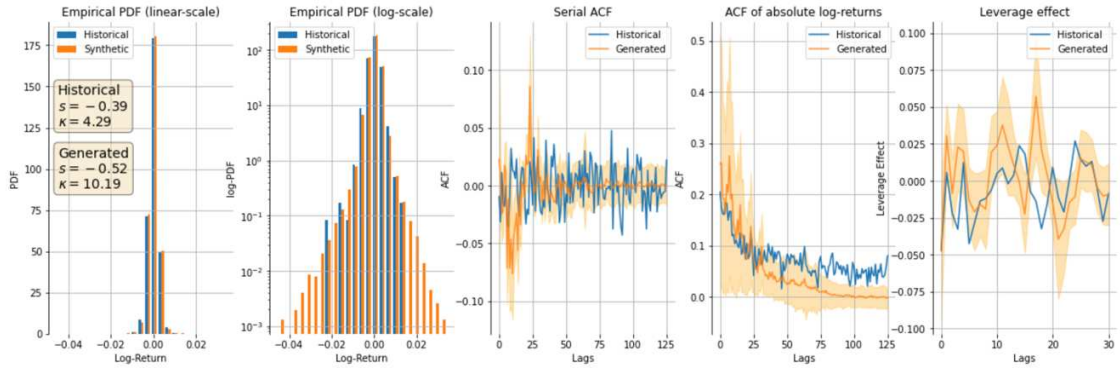

FIGURE 1: Evaluate TCNs path simulator using distributional metrics and dependence scores.

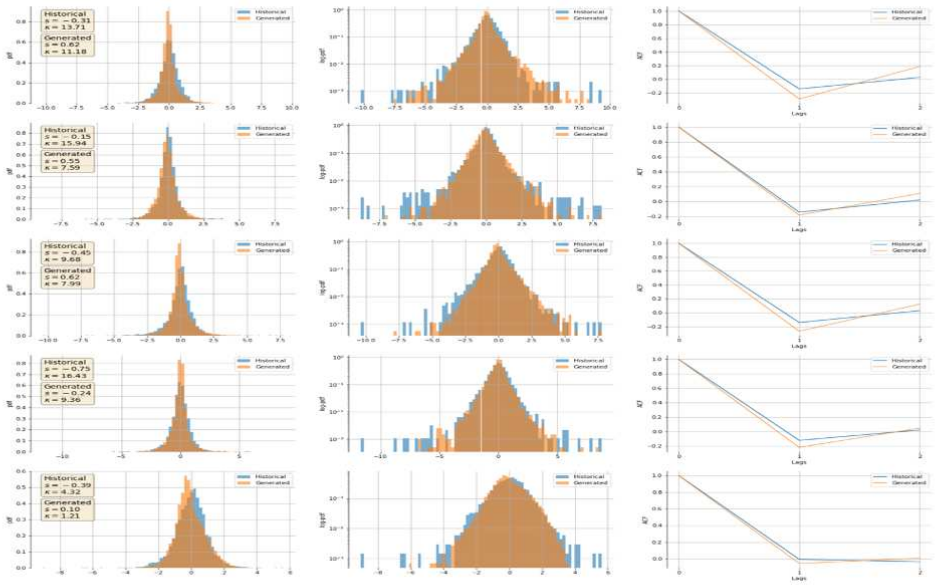

FIGURE 2: Evaluate SigGANS path simulator of US Aggregate Bond Index.

Finally, after examining the feasibility and efficiency of the models we mention above, we conduct a simple portfolio management. The results are shown in Figure 6. We can see that the shape ratio is higher when using the generated time series for portfolio management. This result gives us a plausible application that applying data augmentation for the portfolio management.

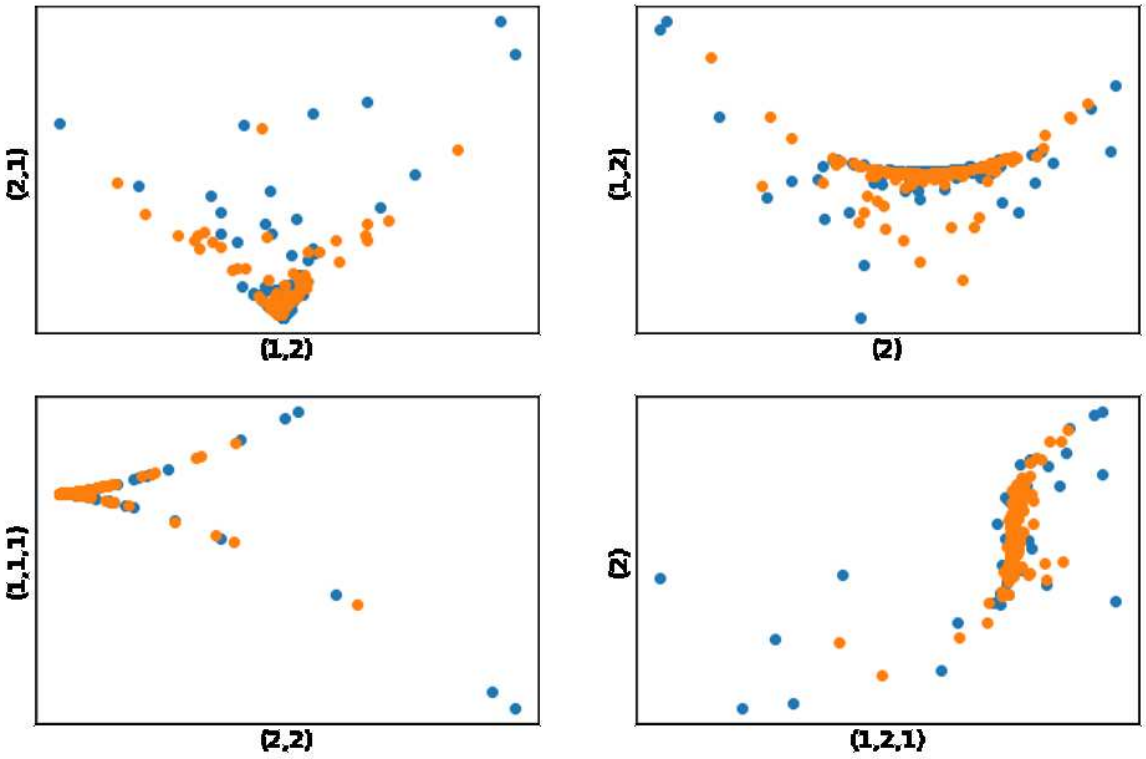

FIGURE 3: VAE projections of US Aggregate Bond Index.

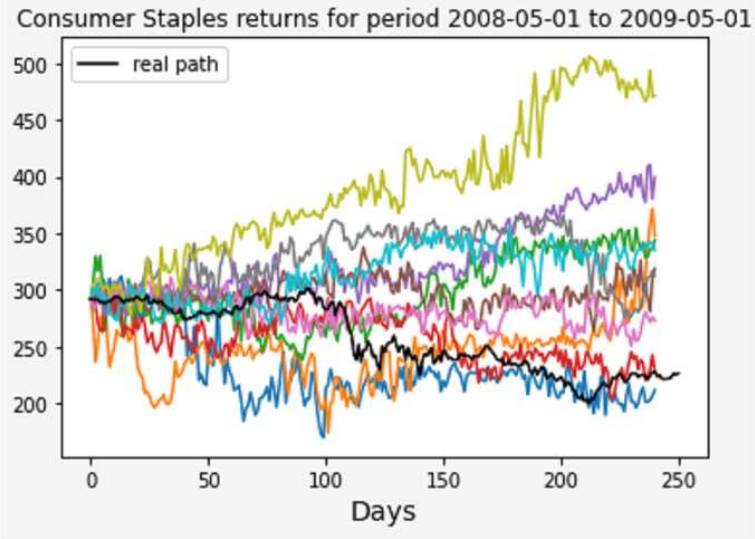

FIGURE 4: Evaluate conditional VAE path simulator of Consumer Staples returns.

5 Conclusion and future work

In this paper, we develop three different machine learning models for financial time series generation. All of the methods have their own advantages and disadvantages, but our work illustrate that it is valid and useful to generated time series through data augmentation

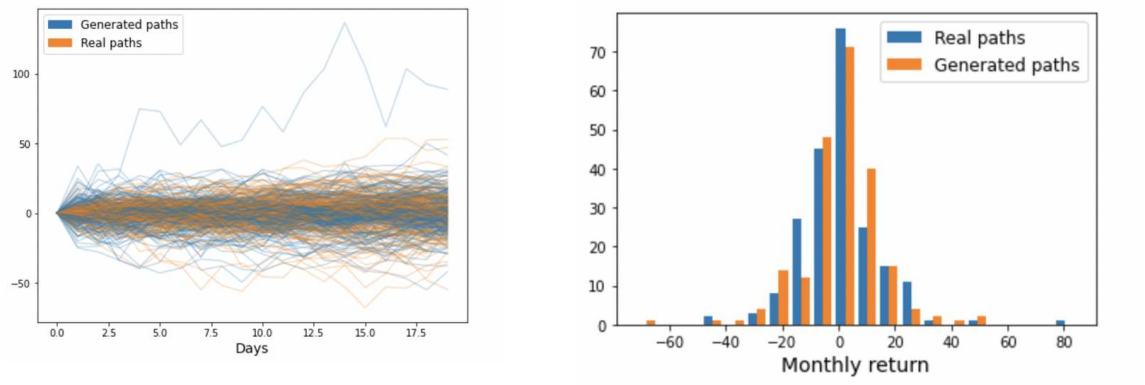

FIGURE 5: VAE generated paths inverted from log-signatures of Consumer Staples returns.

Using Augmented Data	
Expected Annual Return	22.9%
Annual Volatility	16.2%
Sharpe Ratio	1.29

Using Historical Data	
Expected Annual Return	19.2%
Annual Volatility	16.6%
Sharpe Ratio	1.03

FIGURE 6: Compare the return, volatility, and shape ratio between real and simulated data.

algorithms used in machine learning. Moreover, it will make a contribution to risk or portfolio management.

To make those techniques more practical and efficient for applying in the finance field, we will consider conducting scenario analysis using augmented data, generating long-term data, and enhancing computation efficiency of VAE method in future work.

References

[1] Alexei Kondratyev and Christian Schwarz. The market generator. *Available at SSRN 3384948*, 2019.

[2] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[3] Shuntaro Takahashi, Yu Chen, and Kumiko Tanaka-Ishii. Modeling financial time-series with generative adversarial networks. *Physica A: Statistical Mechanics and its Applications*, 527:121261, 2019.

[4] Magnus Wiese, Lianjun Bai, Ben Wood, and Hans Buehler. Deep hedging: learning to simulate equity option markets. *Available at SSRN 3470756*, 2019.

[5] Magnus Wiese, Robert Knobloch, Ralf Korn, and Peter Kretschmer. Quant gans: Deep generation of financial time series. *Quantitative Finance*, pages 1–22, 2020.

[6] Hao Ni, Lukasz Szpruch, Magnus Wiese, Shujian Liao, and Baoren Xiao. Conditional sig-wasserstein gans for time series generation. *arXiv preprint arXiv:2006.05421*, 2020.

[7] Hans Buehler, Blanka Horvath, Terry Lyons, Imanol Perez Arribas, and Ben Wood. A data-driven market simulator for small data environments. *Available at SSRN 3632431*, 2020.

[8] Ben Hambly and Terry Lyons. Uniqueness for the signature of a path of bounded variation and the reduced path group. *Annals of Mathematics*, pages 109–167, 2010.

[9] 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.

[10] Imanol Pérez. Rough path theory and signatures applied to quantitative finance - part 1, 2020.

[11] Ilya Chevyrev, Terry Lyons, et al. Characteristic functions of measures on geometric rough paths. *The Annals of Probability*, 44(6):4049–4082, 2016.

[12] Ilya Chevyrev and Harald Oberhauser. Signature moments to characterize laws of stochastic processes. *arXiv preprint arXiv:1810.10971*, 2018.

[13] Jiawei Chang. *Effective algorithms for inverting the signature of a path*. PhD thesis, University of Oxford, 2018.