Intelligent information systems

UDC 004.7 doi: https://doi.org/10.20998/2522-9052.2023.2.10

Dun Bao[1,2], Oleksandr Zakovorotnyi[1], Nina Kuchuk[1]

[1] National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine
[2] Bank of China, Beijing, China

GENERATING CURRENCY EXCHANGE RATE DATA BASED ON QUANT-GAN MODEL

Abstract. **The aim of the research.** This paper discusses the use of machine learning algorithms to generate data that meets the demands of academia and industry in the context of exchange rate fluctuations. **Research results.** The paper builds a Quant-GAN model using temporal convolutional neural networks (CNN) and trains it on end-of-day and intraday high-frequency rates of currency pairs in the global market. The generated data is evaluated using various statistical methods and is found to effectively simulate the real dataset. Experimental results show that data generated by the model effectively fits statistical characteristics and typical facts of real training datasets with good overall fit. The results provide effective means for global FX market participants to carry out various tasks such as stress tests and scenario simulations. **Future work** includes accumulating data and increasing computing power, optimizing and improving GAN models, and establishing evaluation standards for generating exchange rate price data. As computing power continues to grow, the GAN model's ability to process ultra-large-scale datasets is expected to improve.

Keywords: convolutional neural network; generative adversarial nets, foreign exchange rate.

Introduction

The global foreign exchange market is one of the most active financial markets. Exchange rate fluctuations have a direct impact on market participants and can cause potential risks to their economic status. However, relying solely on historical data for research is not enough. It has become necessary to use methods to generate data that simulates market fluctuations for theoretical and practical verification. In this context, economists and financial traders have adopted various methods to generate financial transaction simulation data. The best results are obtained using artificial intelligence methods [1–6]. The Quant-GAN studied in this paper is an emerging artificial intelligence tool that uses machine learning to generate time-series data of trading prices in financial markets.

1 Generative Adversarial Networks

Generative Adversarial Networks (GAN) is a deep generative model machine learning algorithm that generates realistic data by training a neural network. Goodfellow et al. designed a game-like competition where the generator learns to simulate data that is similar to the real data distribution [7]. The specific algorithm of GAN consists of the following components:

Generator G: Simulates data (images, texts, sounds, time series, etc.) based on certain rules.

Discriminator D: The discriminator D is a classifier that judges (or provides a probability) whether the input data is from the real dataset.

Network training: First, the generator is fixed, and the discriminator is trained using a batch of mixed data that contains both real samples and data generated by the generator. The binary cross-entropy function is used as the loss function, and the discriminator can distinguish between real and fake data after updating the gradient through backpropagation. Second, the discriminator is fixed, and the generator is trained by generating data from random noise inputs. As the discriminator has already been trained in the previous stage, it can identify the authenticity of the input data, and the generator's ability is improved by updating its weight through backpropagation.

Repeat this process so that the generator and the discriminator compete with each other until they reach a Nash equilibrium, completing the training of the entire GAN algorithm. At this point, the data generated by the generator is realistic enough in its distribution, and the discriminator cannot determine whether the input data is real or fake (output a probability of 50%). The optimization problem can be expressed as the solution of Equation:

$$\min_G \max_D V(D,G) =$$
$$= \mathbb{E}_{X\sim p_{data}(x)}[\log D(x)] +$$
$$+\mathbb{E}_{z\sim p_z(z)}\left[\log\left(1-D\big(G(z)\big)\right)\right].$$

It can be proved that in the function space $D(x\ ;\ \theta_d)$ and $G(z\ ;\ \theta_g)$, there is a unique solution that makes G reproduce the training data distribution, at this time $D(x) = 0.5$.

2 Quant-GAN Method

Wiese et al. (2020) proposed the Quantitative Generative Adversarial Network (Quant-GAN) model [8], using the S&P 500 index from 2008 to 2018 as a training set to simulate and generate stock index sequence data. Compared to traditional GARCH, ARIMA, and other machine learning methods, it achieved better results.

Quant-GAN uses uses seven layers of the Temporal Convolutional Networks (CNN) module, based on the Dilated Causal Convolutional Networks architecture [9] proposed by Bai et al. (2018). Each temporal block consists of two one-dimensional expansion causal convolutional layers and two PReLU layers as the activation function. The main feature of CNN is the use of dilated convolution, where the distance moved by the convolution kernel is adjusted by a hyperparameter in each iteration, thus increasing the receptive field without increasing the size of the convolution kernel. At the same time, a skip-connection method for both the generator and discriminator's is also introduced to avoid the gradient disappearance problem. Quant-GAN uses the Wasserstein distance as the loss function, enabling the generated logarithmic return random process to achieve a risk-neutral distribution.

Network training: In the first stage, the generator is fixed, and the discriminator is trained. A batch of real samples and the mixed data generated by the generator are used as inputs. The Wasserstein distance is used as the loss function, and the discriminator updates the gradient through backpropagation to identify real and virtual data. In the second stage, the discriminator trains the generator once every 5 iterations. The discriminator discriminates the generator's randomly inputted data and updates the generator's weights through backpropagation. Repeat the above process for a certain number of times, and the generator and the discriminator compete with each other until they reach the Nash equilibrium, thus completing the training of the entire Quant-GAN. At this point, the generator can generate realistic financial time series data, and the discriminator cannot distinguish between the input data's source whether it is from a real dataset or generated by a generator (with an output probability of 50%).

Data preprocessing: Step 1: Convert the absolute value of the series into logarithmic rate of return: Steps 2 and 4: Standardize (normalize) the log return to a standard normal distribution with mean 0 and variance 1; Step 3: Apply the Inverse Lambert W function transformation to the data; Step 5: For the receptive field T of the discriminator, use a sliding window of the corresponding length to preprocess the logarithmic return sequence.

3 Exchange Rate Data Generation

3.1 Selection, Processing of Data Sets

The foreign exchange transaction dataset used to train the Quant-GAN model comes from the Dukascopy trading platform. The descriptive statistics of the unprocessed original dataset are shown in Table 1.

Table 1 – **Descriptive statistics of preprocessed exchange rate time series data**

currency pair	AUDUSD
date	2019.5.6
type	intraday
amount	4120
maximum value	9.70985
minimum value	9.04364
range	18.75349
average	0
median	0.02582
standard deviation	1
Skewness	0.58207
kurtosis	9.28550

Finally, after steps 1 and 2, the dataset is divided into a training set and a test set according to the ratio of 9:1. used to train the model, and to calculate the fitting error.

3.2 Stylized Facts for Time Series Data in Financial Markets

Aiming at the characteristics of typical financial market time series data represented by stock prices and foreign exchange rates, Cont (2001) summarized some common stylized facts revealed by many studies [10].

Linear Autocorrelation of returns is usually not significant (except for very small intraday time scales), that is, the autocorrelation function converges to near 0 very quickly.

Heavy-tailed distribution refers to the distribution of the return series showing a power-law distribution or Pareto distribution tail characteristics. Leverage effect refers to the price reacting differently to positives and negatives.

Usually, the negative price movement due to bad news is greater. To test the leverage effect, Nelson (1991) used the EGARCH model to estimate the standard deviation of price fluctuations. Asymptotic normality, also known as Aggregational Gaussianity, refers to the phenomenon where the distribution of returns gradually approaches a normal distribution as the time scale increases. Volatility clustering refers to the positive autocorrelation of different volatility measures of a price series within a few time periods. This can be measured with the absolute value correlation coefficient of the return rate.

3.3 Results and Model Evaluation

3.3.1 Data features and model evaluation methods. The exchange rate data generated by the generator after rounds of training is compared with the real dataset. The main features of the recorded data include mean, standard deviation, minimum, quantiles, maximum, skewness and kurtosis. Some typical facts of financial market time series data introduced in the previous section such as autocorrelation, heavy-tailed distribution, leverage effect, aggregational Gaussianity and volatility clustering were tested using both real and generated datasets.

M. Heusel et al. (2017) proposed using Inception Score (IS) and Fréchet Inception Distance (FID) as indicators [11] to evaluate the performance of GAN models. A lower score indicates a more realistic image. According to Xu et al. (2020), this paper trains a KNN classifier to evaluate the performance of the GAN model. When the model is well-trained, the average scores of KNN classifiers calculated using different k values are all around 0.5.

3.3.2 AUDUSD intraday high frequency data. Using the AUDUSD intraday (2019.05.06) high-frequency data, the Quant-GAN model was trained for 500 rounds. The specific parameters are shown in Table 2: Major adjustments include increasing the number of stacked layers in the CNN module to 11 layers in both generator and discriminator and increasing batch size to 2048; increasing stacked layers in CNN module for daytime data to nine layers and increasing batch size to 512. In addition, due to the small absolute value, the learning rate was increased to 0.0002 and gradient clipping value was decreased.

After the model training is completed, random Gaussian noise is used to generate exchange rate logarithmic return data, and the generated data is used to simulate the exchange rate fluctuation path trend (Fig. 1).

Table 2 – **Model hyperparameters**

gradient clipping	0.01
learning rate	0.0002
training rounds	500
batch size	32
sliding window	2048
CNN layers	11

With the increase of training batches, the generator loss stabilizes at 0 and the discriminator loss stabilizes at -0.5 (Fig. 2). The training set and the generated data FID score is stably approaching 0, the test set FID score is approaching the training set (Fig. 2), and the mixed data set KNN classifier discriminant result of generated data and real data is very close to 0.5 (Table 3).

Table 3 – **KNN classifier results**

generator accuracy	0.49891920

The data distribution generated by the model is shown in Fig. 3, and the statistics are shown in Table 4. Compared with the training data set, various statistics such as quantile, mean, standard deviation, skewness, and kurtosis can be effectively fitted.

Table 4 – **Descriptive statistics for real datasets and model-generated data**

	real	**fake**
count	54487	54487
mean	0.00000006	0.00000013
std dev	0.00002522	0.00002310
min	- 0.00057339	- 0.00053218
25%	-0.00001430	-0.00001037
50%	0.00000000	0.00000023
75%	0.00001430	0.00001068
max	0.00060206	0.00039266
skew	-0.34678190	-1.43902695
kurtosis	59.04555456	3 7.37598038

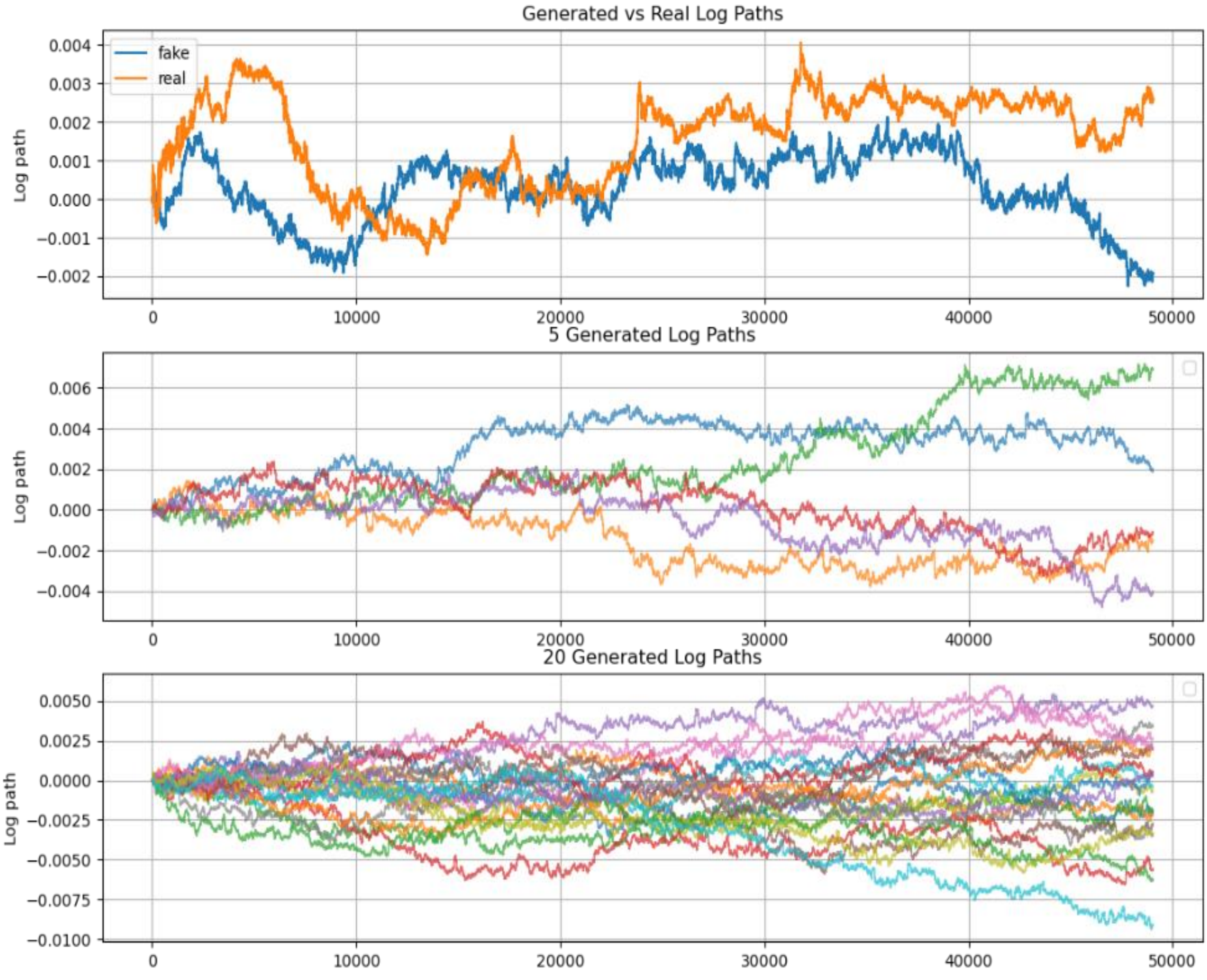

Fig. 1. Generated AUDUSD exchange rate yield trend simulation data

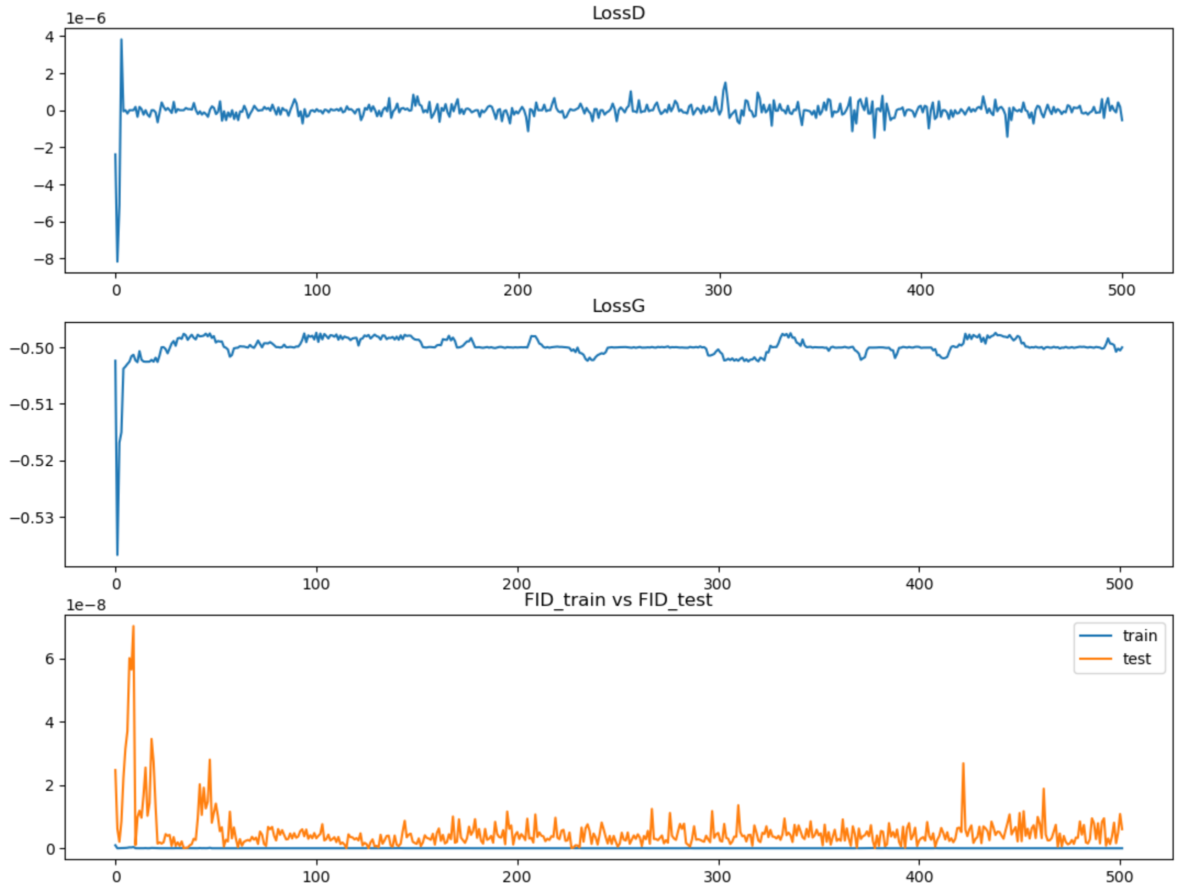

Fig. 2. Quant-GAN model AUDUSD data set learning curve

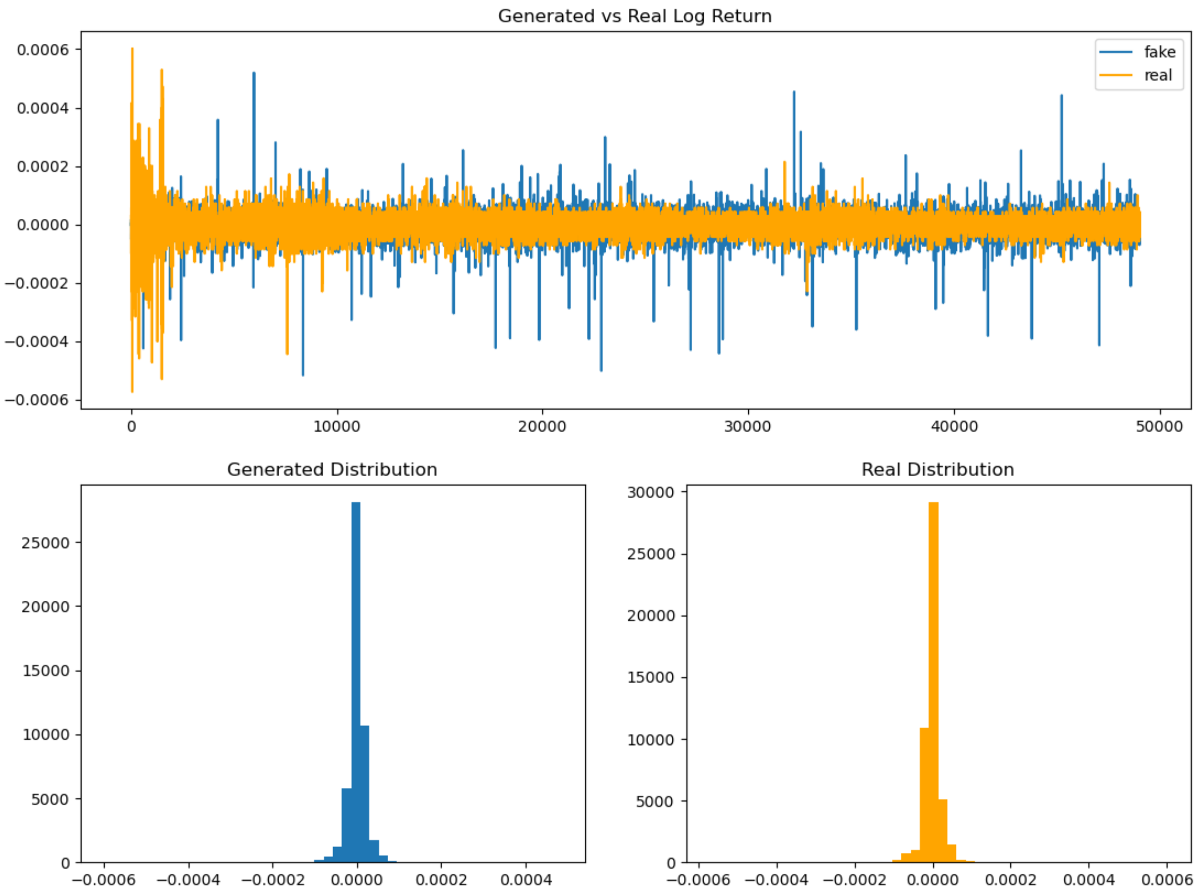

Fig. 3. Generated and real AUDUSD exchange rate logarithmic return distribution

Typical facts of the dataset: as the lag period increases, the autocorrelation between the data generated by the model and the training data is close to 0 (Fig. 4), showing that both lack autocorrelation, the correlation coefficient with the absolute value of the training data set sequence is not 0 and does not decrease as the lag period increases (Fig. 4), indicating an volatility aggregation effect; the tails of the cumulative frequency distribution of the sequence of the training data are far more than those of the normal distribution and the power- law distribution (Fig. 5), indicating that both distributions present heavy tails; the EGARCH parameter estimation results of the model are significant, and the p-value of the coefficient of the asymmetric ARCH term is less than 0.05 (see Table 5), indicating that both sequences have leverage effects.

Various periods of time-lag distribution diagrams show that the two series are gradually approaching the normal distribution as the time scale t increases (Fig. 6).

Results show that the data generated by the Quant-GAN model are in consistent with the training data in terms of main typical facts, such as autocorrelation, distribution heavy tail, leverage effect, aggregational Gaussianity, volatility aggregation, etc.

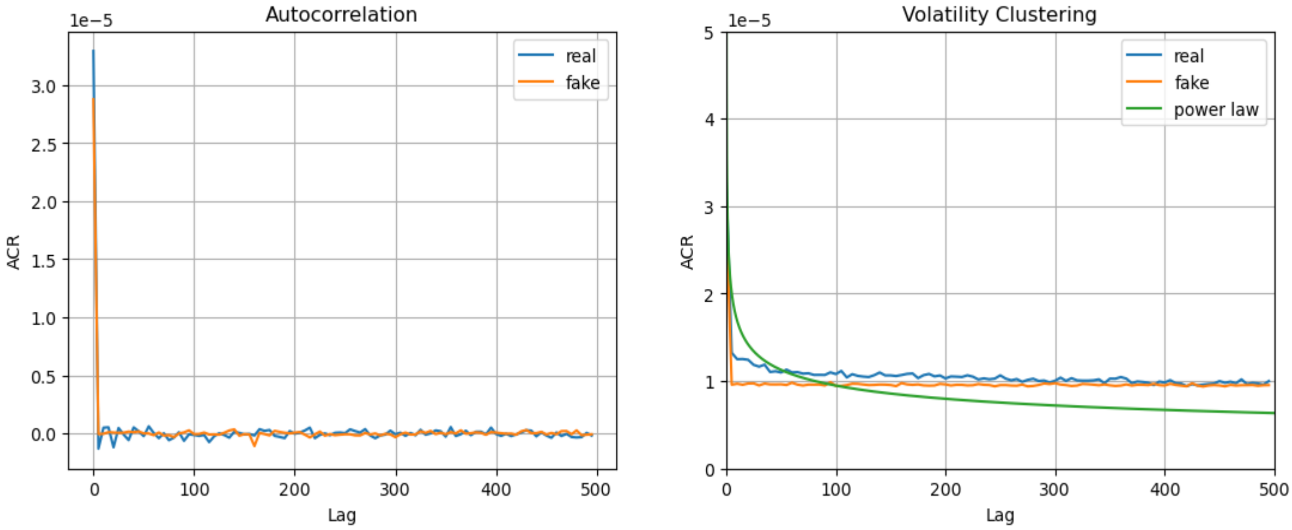

Fig. 4. Generated AUDUSD exchange rate logarithmic rate of return autocorrelation distribution

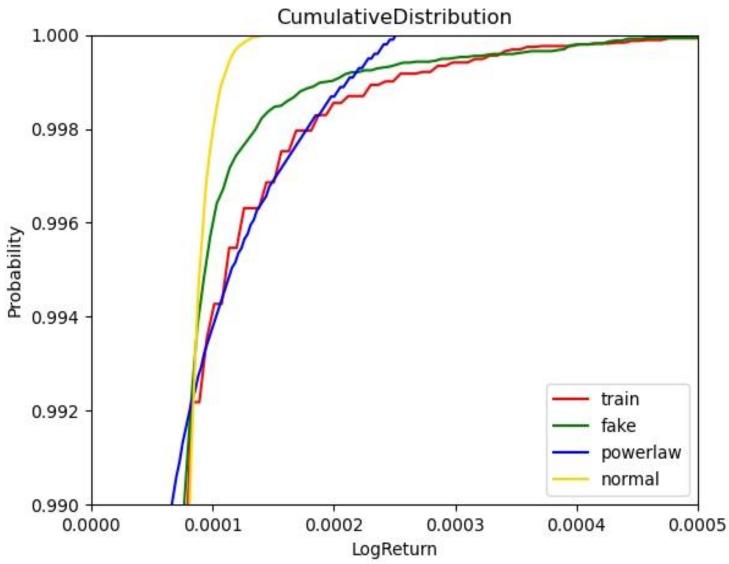

Fig. 5. Generated AUDUSD exchange rate logarithmic return cumulative frequency distribution

Table 5 – **Estimated results of GARCH asymmetric term**

	coeff	std err	t	P>|t|	95.0% Conf. Int.
fake: alpha[1]	0.0935	3.065e-02	3.050	2.289e-03	[3.341e-02, 0.154]
real: alpha[1]	0.0250	6.337e-03	3.952 _	7.751e-05	[1.262e-02, 3.746e-02]

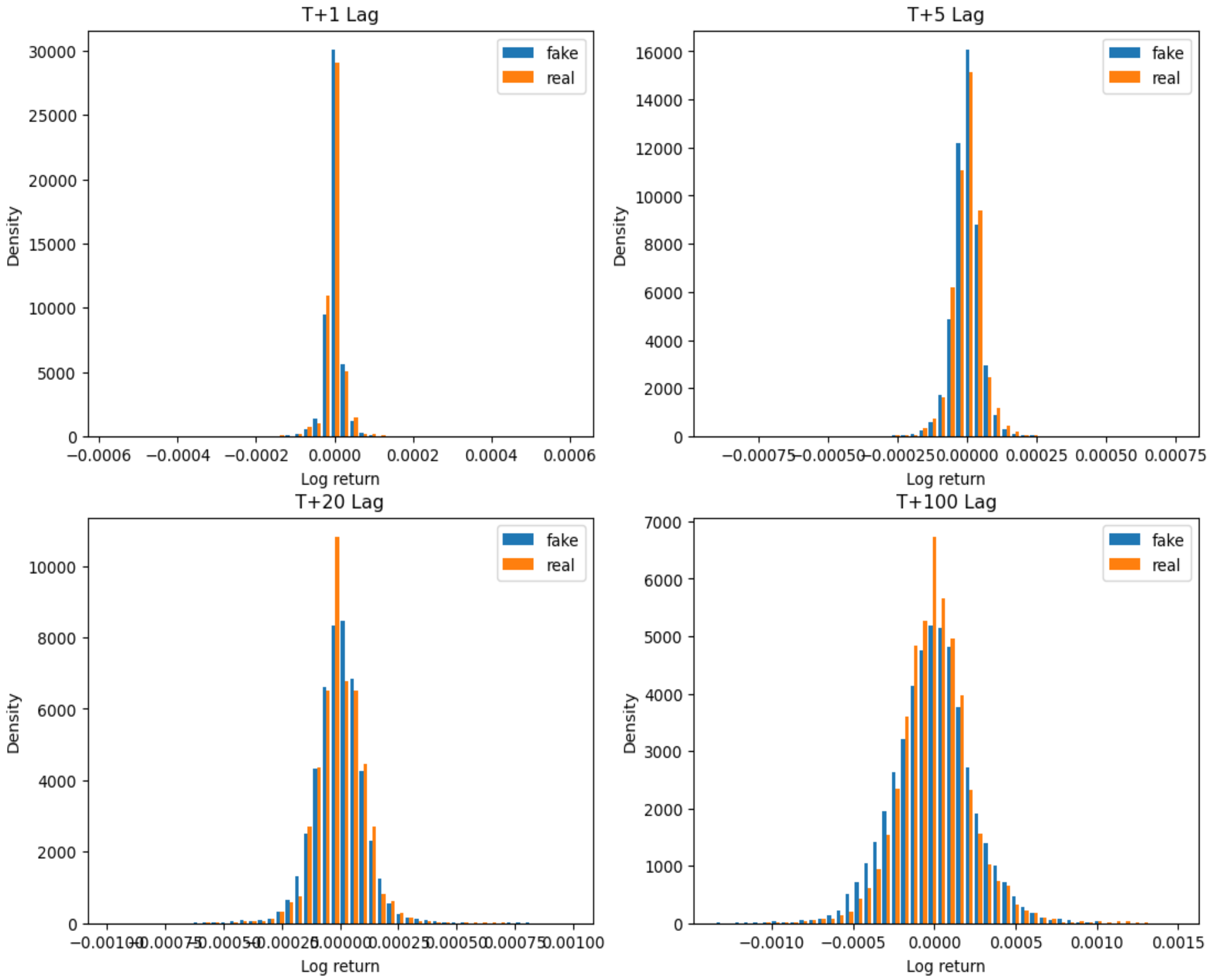

Fig. 6. Distribution of the generated AUDUSD exchange rate logarithmic rate of return

4 Summary and Discussion

This paper introduces the problem of simulating financial market transaction price time series data and related work on machine learning using Generative Adversarial Network (GAN) models. Price data from major currency pairs traded in international markets is selected as the training set to build and train a Quant-GAN model.

The trained model is used to generate exchange rate return rate series data for each currency pair in different time dimensions. Experimental results show that data generated by the model effectively fits statistical characteristics and typical facts of real training datasets with good overall fit.

This research provides global foreign exchange market participants with an effective means of generating exchange rate simulation data for stress testing, scenario simulation, trading strategy and portfolio back-testing, derivatives pricing and more. Applying GAN models to generate and evaluate high-frequency financial time series simulation data expands practical application scenarios for machine learning algorithms such as GAN.

Future work includes accumulating data and increasing computing power, optimizing and improving GAN models, and establishing evaluation standards for generating exchange rate price data. The foreign exchange market is a global over-the-counter (OTC) market where each trader has access to only a small portion of transactions.

Obtaining as much accurate exchange rate transaction data (especially high-frequency data) as possible is essential for model processing. Good data supports good model performance and helps avoid “garbage in, garbage out” situations. Processing massive high-frequency data also requires powerful computing power.

As computing power continues to grow, the GAN model’s ability to process ultra-large-scale datasets is expected to improve.

References

1. Graves, A. (2012), “Supervised sequence labeling with recurrent neural networks”, *Studies in Computational Intelligence*, vol. 385, Springer, Berlin, Heidelberg,142 p., doi: https://link.springer.com/book/10.1007/978-3-642-24797-2.
2. Xu, Q., Gao, H., Yuan, Ya., Guo, Ch., Sun, Yu, Wu, F. and Weinberger, K. (2018), “An empirical study on evaluation metrics of generative adversarial networks”, *arXiv Preprint*, arXiv:1806.07755, doi: https://doi.org/10.48550/arXiv.1806.07755.

3. Kingsep, K. (2021*), Quant gan Code In GitHub*, doi: https://github.com/KseniaKingsep/quantgan.
4. Hlavcheva, D., Yaloveha, V., Podorozhniak, A., Kuchuk, H. (2021), "Comparison of CNNs for Lung Biopsy Images Classification", 2021 IEEE 3rd Ukraine Conference on Electrical and Computer Engineering, UKRCON 2021 – Proceedings, pp. 1–5, DOI: https://doi.org/10.1109/UKRCON53503.2021.9575305.
5. Yaloveha, V., Podorozhniak, A., Kuchuk, H. (2021), "Convolutional neural network hyperparameter optimization applied to land cover classification", *Radioelectronic and Computer Systems*, No. 1(2022), pp. 115–128, DOI: https://doi.org/10.32620/reks.2022.1.09.
6. Podorozhniak, A., Liubchenko, N., Kvochka, M., & Suarez, I. (2021). Usage of intelligent methods for multispectral data processing in the field of environmental monitoring. *Advanced Information Systems*, Vol. *5*, No. 3, pp. 97–102, doi: https://doi.org/10.20998/2522-9052.2021.3.13.
7. Krepych, S., & Spivak, I. (2020). Forecasting system of utilities service costs based on neural network. *Advanced Information Systems*, Vol. *4*, No. 4, pp. 102–108, doi: https://doi.org/10.20998/2522-9052.2020.4.14.
8. Chalyi, S. & Leshchynskyi, V. (2021), "Hierarchical representation of causal relationships to detail explanations in intelligent systems, *Advanced Information Systems*, Vol. *5*, No. 4, pp. 103–108, doi: https://doi.org/10.20998/2522-9052.2021.4.14.
9. 1Goodfellow, I.J, Pouget -Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, Sh., Courville, A. And Bengio, Yo. (2014), "Generative Adversarial Nets", *Advances in Neural Information Processing Systems*, 27, NIPS 2014, Montreal, Canada, doi: https://doi.org/10.48550/arXiv.1406.2661.
10. 2Wiese, M., Knobloch, R., Korn, R. and Kretschmer, P. (2020), "Quant GANs: Deep Generation of Financial Time Series", arXiv Preprint, arXiv: 1907.06673, doi: https://doi.org/10.1080/14697688.2020.1730426.
11. 3Arjovsky, M., Chintala, S. and Bottou, L. (2017), "Wasserstein generative adversarial networks", *International Conference on Machine Learning*, ICML 2017, Sydney, Australia, doi: https://doi.org/10.48550/arXiv.1701.07875.
12. 5Cont. R. (2001), "Empirical properties of asset returns: stylized facts and statistical issues", *Quantitative Finance*, vol. 1 (2), pp. 223-236, doi: http://rama.cont.perso.math.cnrs.fr/pdf/empirical.pdf.
13. 6Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. and Hochreiter S. (2017), "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium", *Advances in Neural Information Processing Systems*, 30, NIPS 2017, Long Beach, United States, doi: https://doi.org/10.48550/arXiv.1706.08500.

Received (Надійшла) 30.03.2023
Accepted for publication (Прийнята до друку) 24.05.2023

ABOUT THE AUTHORS / ВІДОМОСТІ ПРО АВТОРІВ

Дунь Бао – трейдер, Банк Китаю, Пекін, Китай; магістрант кафедри комп'ютерної інженерії та програмування, Національний технічний університет "Харківський політехнічний інститут", Харків, Україна;
Bao Dun – Trader, Bank of China, Beijing, China; master's student of Computer Engineering and Programming Department, National Technical University "Kharkiv Polytechnic Institute", Kharkiv;
e-mail: Bao_Dun@gmail.com; ORCID ID: https://orcid.org/0000-0002-5349-7618.

Заковоротний Олександр Юрійович – доктор технічних наук, професор, завідувач кафедри комп'ютерної інженерії та програмування, Національний технічний університет "Харківський політехнічний інститут", Харків, Україна;
Oleksandr Zakovorotnyi – Doctor of Technical Sciences, Professor, Head of Computer Engineering and Programming Department, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine;
e-mail: Oleksandr.Zakovorotnyi@khpi.edu.ua; ORCID ID: http://orcid.org/0000- 0003-4415-838X.

Кучук Ніна Георгіївна – доктор технічних наук, професор, професор обчислювальної техніки та програмування, Національний технічний університет "Харківський політехнічний інститут", Харків, Україна;
Nina Kuchuk – Doctor of Technical Sciences, Professor, Professor of Computer Science and Programming Department, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine;
e-mail: nina_kuchuk@ukr.net; ORCID ID: http://orcid.org/0000-0002-0784-1465.

Генерація даних про обмінний курс валюти на основі моделі Quant-GAN

Дунь Бао, О. Ю. Заковоротний, Н. Г. Кучук

Анотація. Мета дослідження. У цьому документі обговорюється використання алгоритмів машинного навчання для генерації даних, які відповідають вимогам наукових кіл та промисловості в контексті коливань обмінного курсу. **Результати дослідження.** У статті створено модель Quant-GAN з використанням часових згорткових нейронних мереж (CNN) і навчено її на високочастотних курсах валютних пар на світовому ринку наприкінці дня та всередині дня. Згенеровані дані оцінюються за допомогою різних статистичних методів і виявляються такими, що ефективно імітують реальний набір даних. Експериментальні результати показують, що дані, згенеровані моделлю, ефективно відповідають статистичним характеристикам і типовим фактам реальних навчальних наборів даних із загальною хорошою відповідністю. Результати надають учасникам глобального валютного ринку ефективні засоби для виконання різноманітних завдань, таких як стрес-тести та моделювання сценаріїв. **Майбутня робота** включає накопичення даних і збільшення обчислювальної потужності, оптимізацію та вдосконалення моделей GAN, а також встановлення стандартів оцінки для генерації даних про ціни обмінного курсу. Оскільки обчислювальна потужність продовжує зростати, очікується, що здатність моделі GAN обробляти надвеликі масиви даних покращиться.

Ключові слова: згорткова нейронна мережа; генеративні змагальні мережі; валютний курс.