Digital Finance (2023) 5:29–56
https://doi.org/10.1007/s42521-022-00050-0

ORIGINAL ARTICLE

DeepVaR: a framework for portfolio risk assessment leveraging probabilistic deep neural networks

Georgios Fatouros[1,2] · Georgios Makridis[1] · Dimitrios Kotios[1] · John Soldatos[2] · Michael Filippakis[1] · Dimosthenis Kyriazis[1]

Received: 19 July 2021 / Accepted: 21 March 2022 / Published online: 13 April 2022
© The Author(s) 2022

Abstract

Determining and minimizing risk exposure pose one of the biggest challenges in the financial industry as an environment with multiple factors that affect (non-)identified risks and the corresponding decisions. Various estimation metrics are utilized towards robust and efficient risk management frameworks, with the most prevalent among them being the Value at Risk (VaR). VaR is a valuable risk-assessment approach, which offers traders, investors, and financial institutions information regarding risk estimations and potential investment insights. VaR has been adopted by the financial industry for decades, but the generated predictions lack efficiency in times of economic turmoil such as the 2008 global financial crisis and the COVID-19 pandemic, which in turn affects the respective decisions. To address this challenge, a variety of well-established variations of VaR models are exploited by the financial community, including data-driven and data analytics models. In this context, this paper introduces a probabilistic deep learning approach, leveraging time-series forecasting techniques with high potential of monitoring the risk of a given portfolio in a quite efficient way. The proposed approach has been evaluated and compared to the most prominent methods of VaR calculation, yielding promising results for VaR 99% for forex-based portfolios.

Keywords Probabilistic deep neural networks · Time-series · Forex · Finance · VaR · Risk assessment · VaR prediction

JEL Classification C22 · C45 · C53 · C63

✉ Georgios Fatouros
gfatouros@unipi.gr; gfatouros@innov-acts.com

Extended author information available on the last page of the article

1 Introduction

Risk assessment in the financial sector has drawn considerable attention in recent years, as the extensive use of Value at Risk (VaR) among other risk management models proved to be inefficient in measuring the financial risk during recession periods, such as at the latest global financial crisis of 2008 and the 2020 COVID-19 pandemic. VaR, being originally designed for internal use in financial institutions, has become a key factor in determining capital requirements and conducting risk assessment, especially after the introduction of the Basel II Accord (De Waal et al. 2013). As a result of stricter regulations regarding capital management, alongside technological advances utilized in the financial sector, many institutions are actively exploring and promoting more accurate methods of monitoring and assessing the exposure to different risks (Elsinger et al. 2006; Zhao 2020).

Although VaR has caused high controversy throughout the years, it is considered to be a widely used internationally instituted financial risk model among several other financial tools and both quantitative analysts and regulatory authorities started paying more attention to VaR, since its standardization as a risk measure in 1996 by RiskMetrics (Longerstaey and Spencer 1996). The core of the criticism for VaR models derives from their simplistic assumptions and their insufficient results during periods of crises (Abad et al. 2014). The controversy raised over the VaR model can be summed in the indicative phrase of Einhorn and Brown in 2008, where VaR is compared to “an airbag that works all the time, except when you have a car accident” (Einhorn and Brown 2008).

The notion of VaR reflects the maximum expected loss of a portfolio over a given time horizon, at a predefined confidence level. Given the fact that VaR, besides its extensive use in the financial sector, has also been extensively studied in the academic literature, this paper does not further elaborate on the VaR definition. However, as several variants of VaR are emerging from the financial engineering and digital finance literature, we focus on the three major VaR categories, namely the non-parametric, the parametric, and the semi-parametric.

In the non-parametric models, assumptions regarding the distribution of the returns of the underlying portfolios are not required. The main advantage of this type of method is the low computational complexity. The Historical Simulation (HS) is the main representative of this category, where the empirical distribution of past portfolio returns is used to calculate VaR. The latter can be obtained by taking the required quantile of this distribution for a given history-window. The main disadvantage of this method is that it fails to capture unseen fluctuations that are not present in the utilized history-window (Chang et al. 2003).

Contrastingly, in parametric approaches, a valid model of the portfolio returns and their distribution should be theoretically defined prior to VaR estimation. On the one hand, most of the parametric models are simple in terms of implementation when Gaussian or Student-t distribution is assumed (Abad et al. 2014). On the other hand, this assumption does not hold for most of the financial time-series data. Some well-known methods falling in this category are the

Variance–Covariance method (VC) and many GARCH-variants methods. For example, when VC method is applied assuming Gaussian distribution, a proper history-window should be defined, and the variance–covariance matrix of returns should be calculated, then the VaR can be obtained by Eq. 1

$$\mathrm{VaR}^{a} = z_{1-a}\sqrt{W^{T}\sum W}, \tag{1}$$

where a, z, and w are the confidence probability of VaR (i.e., 99%), the z-score, and the portfolio weights, respectively.

Finally, the semi-parametric VaR models are produced by combining the two aforementioned methods, where some assumptions are made either about the "error" distributions, its extremes, or the model dynamics. The Monte Carlo method (MC) is the main semi-parametric method, where scenarios are randomly generated for future returns of the portfolio, based on some non-linear pricing models. MC is more reliable comparing to historical and parametric approaches when dealing with complex portfolios and complicated risk factors. Its core assumption is that the risk factor has a known probability distribution, i.e., that market factors follow certain stochastic processes, which are used to estimate future returns (Abad et al. 2014).

Based on pertinent literature (Angelidis and Degiannakis 2018; Yamai and Yoshiba 2002), most of the existing VaR methods are facing various challenges. The most critical challenge is the severe VaR violations during which the portfolio realizes a loss exceeding the VaR value due to dependencies between the VaR predictions, especially for 99% confidence level. This is often the case when high market downturns occur. In addition, the high excess loss, beyond the VaR threshold that happens due to the fat tails of the financial time-series distribution and the leverage effect, is rarely taken into account. It should be also noted that despite the fact that there are several approaches providing a risk assessment management framework, the majority of related published literature examines single asset "portfolios" (i.e., S&P index). These drawbacks of the VaR methods reflect additional motivations for the approach presented in this paper.

The financial institutions started using VaR as a risk estimation metric to ensure their survival during catastrophic events, after the stock market crash on Wall Street in October 1987. The fact that the 2008 financial crisis resulted in an overall loss of $3.4 trillion among all major financial institutions over the world according to the International Monetary Fund (IMF) (Dattels and Miyajima 2009) is an example of the need of an efficient (innovative) VaR prediction methodology. Thus, it is crucial now that the COVID-19 pandemic has affected the global economy to a great extent (Das et al. 2021) to revise the risk-assessment tools and address their methodological limitations.

To this end, the research should be conducted in the context of portfolios consisted of certain type of financial assets (i.e., forex, bonds, and stocks). Given that VaR is independent of the type of assets comprising the portfolio and that there is abundance of open-source data for the majority of the forex (FX) instruments, portfolios based on FX assets were opted for.

This paper introduces a data-driven framework that predicts portfolios' VaR, addressing the above-mentioned challenges with the following key innovations:

1. Integrates a continuous learning approach that considers the latest market prices avoiding clustered VaR violations and thus addressing the dynamic nature of financial data.
2. Is able to capture rare market events with very short training time by utilizing probabilistic forecasting based on auto-regressive recurrent neural networks in the context of VaR.
3. Goes beyond single-asset pre-trade/what-if analysis (i.e., asset-level) to portfolio pre-trade/what-if analysis (i.e., portfolio-level). It should be noted that this is achieved in (near) real time by eliminating the need to re-train the neural network model.

Even though considerable emphasis is given on portfolios composed of FX assets and daily VaR, the proposed approach is also applicable to other types of financial instruments and different time horizons, since it could be applied and optimized for various types of time-series. Moreover, the proposed framework has been evaluated with several loss functions and two different coverage tests.

This paper is structured as follows: Sect. 2 presents the related work in the relevant fields of study, while Sect. 3 describes the proposed methodological approach delivering details regarding the data and the evaluation scheme utilized. Sections 4 and 5 present the results of the back-testing to demonstrate and evaluate the performance and effectiveness of the proposed mechanisms. Finally, Sect. 6 closes the paper with recommendations on future research and further potentials of the current study.

2 Related work

The foundations of VaR in finance as a risk-assessment approach were introduced in 1996 by Morgan and Reuters (Longerstaey and Spencer 1996). This parametric method became dominant in the financial sector and has been extensively utilized under the name "RiskMetrics model", despite the fact it suffers from unrealistic assumptions such as the normally, independently, and identically distributed financial returns.

The aforementioned limitations, combined with the highly competitive and demanding nature of the financial markets, moved the research from the parametric method, to alternative independent directions. There are various studies comparing/back-testing different VaR approaches, with the one presented in Kuester et al. (2006) standing out. Summarizing its results, all the unconditional models produce clustered VaR violations, yet some may still pass as acceptable when considering only the (unconditional) violation frequencies. Though, this conclusion depends to some extent on the chosen window size, with less-parameterized models having an advantage as history-window size decreases from 1000 to 250. On the other hand, conditional VaR models lead to much more volatile VaR predictions and may arguably cause problems in allocating capital for trading purposes. Additionally, as discussed in Abad et al. (2014), most of the researchers focused on the family of GARCH models, arguing also that asymmetric GARCH models yield better results, but without statistically significant difference. Equally effective are many VaR applications, that utilize the Extreme Value Theory (EVT) approach. The key-characteristic of EVT is that it focuses on limiting the distribution of extreme returns observed over a long time-period, which is essentially independent of the distribution of the returns. There is an indicative comparative evaluation of the predictive performance of various VaR models, emphasizing on the two EVT-based methodologies, Peak Over Threshold model (POT) (Novak 2011), and block maxima model (BM) (Mcneil 1998). Its results demonstrated that, although some traditional methods might yield comparable results at the conventional confidence levels, the EVT methodology produces the most accurate forecasts of extreme losses for high confidence levels (Bekiros and Georgoutsos 2005).

Despite the widespread use of various econometric models regarding VaR estimation and modeling of financial returns, the rise of machine learning and deep learning models offered an improved toolkit to financial firms, introducing innovative and more effective approaches and automating many financial tasks. In the last few years, several studies have been carried out, such as (Lim and Zohren 2020; Sen et al. 2019), analyzing how some of the most prominent deep learning architectures [such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) (Gers et al. 2000)] can be used for time-series forecasting. Such models have been widely applied across different domains, due to their ability to model non-linear temporal patterns. For instance, in Neuneier (1996) and Xiong et al. (2018), a neural network-based reinforcement learning model is used to perform portfolio management. In Weng et al. (2017), an ensemble model of machine learning algorithms has been utilized to predict stock market movement, proving that an LSTM neural network has a high potential for predicting financial time-series. Moreover, a Generative Adversarial Networks (GAN)-based model (Goodfellow et al. 2014) has been proposed to generate synthetic representative financial data sets, demonstrating that the synthetic distributions share similar characteristics with the real data (Pfenninger et al. 2021). Finally, a combination of wavelet analysis and an LSTM neural network enables to capture the complex features of financial time-series, such as non-linearity, non-stationarity, and sequence correlation (Yan and Ouyang 2018).

As far as the VaR estimation is concerned, two key challenges arise from the relative literature: (i) The first one refers to the excess loss of VaR violations, which is rarely taken into account. For instance, in Hendricks (1996), the authors argued that, although a 99% risk measure may sound as if it is capturing essentially all of the relevant events, the other 1% of events can in extreme cases entail losses substantially in excess of the risk measures generated on a daily basis. (ii) The second challenge is that most of VaR 99% models produce more violations than the nominal confidence probability, with limited back-testing regarding the coverage and independence of VaR estimations. Moreover, most of the related research papers focus on certain financial indexes (i.e., S&P) as a univariate time-series, without taking into account the correlations among the assets that comprise the examined portfolio.

Based on the above, the main contribution of the current research is twofold. First, a novel probabilistic approach for VaR prediction is proposed based on DeepAR, providing the full predictive distribution and allowing decision-makers to optimize their actions. This approach is performing better in VaR 99% than the most prevalent VaR estimation methods. Second, evaluation/back-testing procedures based not only on univariate assets but also on portfolios are presented. Furthermore, various evaluation metrics and statistical tests were utilized to illustrate the efficiency of the proposed approach.

3 Proposed approach

An innovative framework for portfolio VaR estimation is proposed, utilizing probabilistic deep neural networks (Mohebali et al. 2020). However, regularized datasets are not available, as hardly any hedge fund or individual trader is willing to disclose proprietary information of their portfolios, while in most of the relevant studies only a univariate index (like S&P) is leveraged as a portfolio. In our approach, the VaR of each asset (VaR_i) is initially calculated, then the portfolio's VaR (VaR_p) is derived from the combination of VaR_i, taking into account their correlations and their corresponding weights. Several random portfolios have been created to evaluate our results for different cases and scenarios.

Furthermore, to evaluate the proposed model, five different VaR methods were explored, namely, GARCH, RiskMetrics (RM), HS, BiGAN, and MC. The evaluation was based on several loss metrics, such as the number of VaR violations, quadratic loss and firm loss, and coverage tests described in Sect. 3.2.3.

In our approach, both VaR and Profit and Loss (PnL) are expressed in log-return terms, while the terms PnL and portfolio returns r are used interchangeably in the remainder of this paper. Mathematically, let $\text{PnL}_t = r_t = \log \frac{P_t}{P_{t-1}}$ be the log-returns, and P_t is the close price of the financial instrument on day t. The 1-day VaR on day t is defined as

$$P(r_t \leq \text{VaR}_t^{\alpha}) = 1 - \alpha. \tag{2}$$

As shown in Eq. 2, VaR is expressed in return terms, thus given a distribution of return, VaR can then be determined and expressed in terms of a percentile of the return distribution (Christoffersen et al. 2001). Specifically, if q_α is the α-th percentile of the continuously compounded return, then VaR can be expressed as

$$\text{VaR}^a = q_{1-a}. \tag{3}$$

3.1 DeepVaR

The approach presented in this paper exploits deep neural networks and probabilistic forecasting. Having tested different probabilistic time-series forecasting models, DeepAR Estimator as described in Salinas et al. (2020) has been selected as the core of the proposed framework, the so-called DeepVaR.

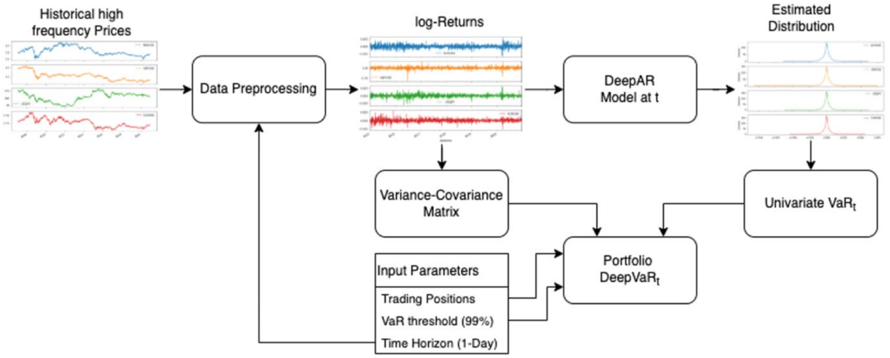


Fig. 1 Conceptual architecture of DeepVaR framework

DeepAR is based on an auto-regressive recurrent neural network model, specifically designed for multivariate time-series modeling producing accurate probabilistic forecasts. It models the conditional distribution $P(z_{i,t_0:T} \mid z_{i,1:t_0-1})$ of the time-series z_i for future time-steps from t_0 to T given the past values of the z_i from time-step 1 to $t_0 - 1$. It is assumed that the model distribution $Q_\theta(z_{i,t_0:T} \mid z_{i,1:t_0-1})$ consists of a product of likelihood factors. The latter are maximized to train a deep neural network (RNN) that learns the distributions' parameters. Several samples can be easily generated from the estimated distribution in an MC fashion. Moreover, this algorithm can be trained with several similar time-series simultaneously, enabling cross-learning between them.

This algorithm has been applied successfully in several business sectors and real-life scenarios, such as food safety (Makridis et al. 2020), where a set of probabilistic techniques were introduced to provide insights regarding potential food recalls, in retail sector predicting the number of sales per product (Khan et al. 2020) and in industry estimating electricity demand (Wu et al. 2020). This innovative algorithm has been complemented with additional features as part of the overall framework presented in this paper. These features (such as the portfolio-level predictions and the continuous learning) were deemed essential to successfully utilize DeepVaR in the finance sector. The framework predicts VaR of FX portfolios, where the individual time-series share similar dependencies and the overall goal is to draw the portfolio's returns distribution.

Furthermore, DeepVaR proposes a continuous learning approach instead of a classical machine learning pipeline where the model is trained once and for several hours in a large dataset. Contrastingly, in DeepVaR, the model is retrained on the latest market data, addressing the dynamic nature of financial data while avoiding at the same time model bias and drift, serial correlation between VaR estimations and clustered VaR violations (Mehrabi et al. 2021). To this end, DeepAR parameters were optimized to enable model training in a timely manner (< 13 s), making it applicable even for intra-day VaR estimations.

As depicted in Fig. 1 and further analyzed in terms of sequence flows by Algorithm 1, the DeepVaR framework performs estimations on asset-level (i.e., for a single VaR) at each time-step t. Historical market prices x_i from t_0 to $t-1$ of multiple instruments i are ingested into the framework simultaneously. During the data preprocessing step, the input data are initially resampled to match the frequency of the selected VaR time horizon and then are transformed to log-returns ($r_{i,t}$). For example, in the case of minute data and daily VaR selection, the input time-series is transformed to daily log-returns. The latter is used to train the DeepAR model and to estimate the distribution of each time-series (i.e., asset-level) for time-step t. With the distribution of the assets' returns available, $\text{VaR}_{i,t}$ can be obtained from Eq. 3. In the last step, portfolio-level predictions, i.e., portfolio $\text{VaR}_{p,t}$ is estimated based on the returns variance–covariance matrix, $\text{VaR}_{i,t}$ and the input weight on each asset (see Sect. 5 for more details). It is also noted that the process of calculating VaR_p from VaR_i requires only matrices multiplications (see Eq. 15). Thus, no training is required and, therefore, the overall process is quite time-efficient. Thus, DeepVaR could be also used for what-if analysis comparing portfolios' risk against the different weights on input assets/instruments.

Algorithm 1 VaR prediction using the DeepVaR framework

```
Input
    $X$         Historical Prices $\in R^{T\times N}$
    $w$         Portfolio weights $\in R^{1\times N}$
    $freq$      Time horizon of VaR
    $\alpha$    VaR confidence probability
Output
    $VaR_p^\alpha$   Portfolio VaR estimation
Data Preprocessing
$prices \Leftarrow$ RESAMPLE$(X, freq)$
$returns \Leftarrow \log(1 +$ PCT_CHANGE$(prices))$
$returns \Leftarrow$ DROPNA$(returns)$
$train\_ds \Leftarrow returns[-900:,:]$ comment: training dataset
$R \Leftarrow$ COV$(train\_ds[-125:,:])$ comment: variance covariance matrix
Model training and forecasting
$model \Leftarrow$ DEEPARESTIMATOR$(predictionLength = 1,$
$contextLength = 15,$
$freq = freq,$
$numLayers = 2,$
$dropoutRrate = 0.1,$
$cellType = "lstm",$
$numCells = 50,$
$trainer = Trainer(epochs = 5, lr = 0.0001, numBatchesPerEpoch =$
$50))$
$estimator \Leftarrow$ TRAIN$(model, train\_ds)$
$pred \Leftarrow$ PREDICT$(estimator, num\_samples = 1000)$
Calculate portfolio $VaR_p^\alpha$
$lower\_q \Leftarrow$ QUANTILE$(pred, q = 1 - \alpha, axis = 0)$
$upper\_q \Leftarrow$ QUANTILE$(pred, q = \alpha, axis = 0)$
$V \Leftarrow$ ARRAY$([1, 4])$ comment: initialize empty array
for $i$ in RANGE$(N)$ do
    if $w[i] < 0$ then
        $V[i] \Leftarrow w[i] \times lower\_q[i]$
    else
        $V[i] \Leftarrow w[i] \times upper\_q[i]$
    end if
end for
$VaR_p^\alpha \Leftarrow -$SQRT$(V \times R \times V^T)$
```

Additionally, due to noisy data, the initialization of the weights of the RNN model, which is random due to the stochastic nature of this optimization process, had quite an impact on the performance. This pitfall is common when noisy data are concerned as multiple local minima in the loss surface exist. This is quite challenging as the loss surfaces are generally non-convex and may have multiple saddle points making it difficult to achieve model stability. To tackle this challenge, different models (10 in total) with different random seeds (i.e., initial weights) were trained and ensembled as a Bagging model, as proposed by Montavon et al. (2012). All these models share the same hyper-parameters such as 900 observations training set, 5 epochs, 1-day prediction length, and 15-day context length, and were fine-tuned using grid-search technique and intuition, as a fundamental step of the machine learning pipeline. Moreover, the selected model features the ADAM optimizer (Kingma and Ba 2014) with the learning rate set to 0.0001, 2 LSTM layers with 50 cells each, and dropout 10%.

3.2 Evaluation approach

In this section, a brief description of the employed baseline models is initially given, and then, the utilized data along with the portfolio contraction procedure are analyzed. Considerable emphasis is also given on both the evaluation metrics and the statistical tests that enable the comparison between several VaR models.

3.2.1 Baseline techniques

Towards evaluating the proposed DeepVaR framework, a comparison against the four most used VaR estimation techniques and one deep learning model was performed. A short description of these techniques follows.

GARCH type models have been widely used for VaR estimation, producing considerably well results in exchange rate data (So and Philip 2006). Such models can capture the time-varying volatility feature characterizing financial time-series. Thus, a GARCH model can be utilized to predict the future volatility of the returns. Since GARCH(1,1) model was found to be adequate to many financial time-series (Bollerslev et al. 1992), we have chosen it as one of the three baseline models, which can be described as

$$r_t = \mu + \epsilon_t, \tag{4}$$

$$\epsilon_t = \sigma_t e_t, e_t \sim N(0,1), \tag{5}$$

$$\sigma_t^2 = \omega + \alpha \epsilon_{t-1}^2 + \beta \sigma_{t-1}^2. \tag{6}$$

Future returns r_{t+1} can be obtained by estimation of the parameters ω, α, β in Eq. 6 using the maximum-likelihood method. To have comparable results to the DeepVaR, the parameters of GARCH(1, 1) model are estimated over a sample of 900 observations.

The second baseline technique refers to the RM model for VaR estimation. As proposed in Longerstaey and Spencer (1996), it is a GARCH(1,1) variant, modeling the volatility of returns at the next time-step as exponential weighted moving average of the past volatilities $\sigma_t^2 = \lambda\sigma_{t-1|t-2}^2 + r_{t-1}^2(1-\lambda)$, $\lambda = 0.94$, while the mean of returns μ considered is equal to zero. The value of the decay factor λ implies that the effective historical data to forecast future volatility are approximately 74 days.

The third baseline technique, HS, uses a configurable number of past historical observations to calculate the portfolio's the actual percentiles of this observation period as value-at-risk measures. For example, for an observation period of 1000 days, the 99th percentile HS value-at-risk measure is the 11th largest loss observed in the sample of 1000 outcomes (because the 1 percent of the sample that should exceed the risk measure equates to 10 losses) (Hendricks 1996). The effective historical window for the HS model was fine-tuned to 1000 days.

In addition, an MC-based VaR model has been employed to evaluate the performance of the proposed solution, which also includes an MC procedure. The MC model produces random samples from the normal distribution to estimate the future distribution of the portfolio returns. The generated distribution is then used to calculate VaR. The input parameters (i.e., μ, σ^2) for the MC model have been calculated from the last 900 historical data that offered better performance in comparison to a lower number of historical observations.

Another baseline technique baseline technique leverages Bidirectional Generative Adversarial Networks (BiGAN) (Donahue et al. 2016) towards modeling the joint probability distribution of the portfolio returns without the need to specify their distribution explicitly. The BiGAN consists of three neural networks, a generator (G) that learns to produce realistic synthetic samples from the latent space z, an encoder (E) which learns to map data x to latent representations z, and a discriminator (D) distinguishing jointly real from synthetic samples and real from synthetic encodings. After the BiGAN training, the generator is used to produce samples that will be close to the actual returns' distribution, and based on them, VaR is obtained. The utilized BiGAN, based on the code available in GitHub,[1] was evaluated following the continuous learning approach (i.e., retraining the model at each new data observation) introduced by the DeepVaR framework. To have comparable results to the DeepVaR, the selected training data size was set to 900 data points, while 75 epochs were required to calibrate the binary cross-entropy loss of BiGAN components.

3.2.2 Dataset description

In the context of the evaluation of the proposed approach against the baseline techniques, the four FX instruments (AUDUSD, GBPUSD, USDJPY, and EURUSD) with the highest liquidity among the rest have been chosen as the underlying dataset. Specifically, the dataset consists of daily close prices, ranging from 2007/01/01 to 2020/12/18, with the data being obtained from http://www.eatradingacademy.com. The daily prices, represented as time-series, were transformed to log-returns to make them stationary, while the VaR predictions are measured in this scale.

First, the 1-day VaR of each FX pair is calculated from 2018/01/01 to 2020/12/18 in a rolling window prediction format with this time period serving also as test

[1] https://github.com/hamaadshah/market_risk_gan_tensorflow.

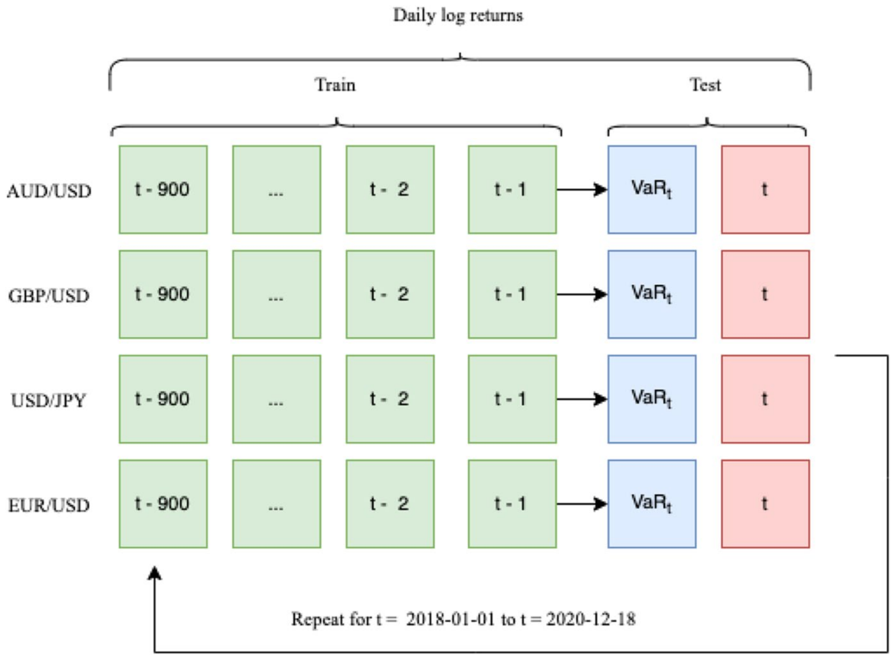


Fig. 2 Training dataset for rolling window VaR estimation

dataset for the models' evaluation. The training window, for the DeepVaR, BiGAN, and GARCH models, was set to 900 data points consisting of the latest 900 daily log-returns of each FX asset (see also Fig. 2). For the HS model, the historic window was fine-tuned to 900 days, contrary to RiskMetrics, where only the last 74 days were taken into account.

Second, to increase the reliability of our results, 1000 portfolios have been created reflecting the historical behavior of 1000 different traders. The asset allocation in each of these portfolios was generated randomly and expressed as different proportions of the four FX instruments. These portfolios may include both long and short positions (i.e., positive and negative weighting coefficients), while the sum of the absolute weights is equal to one. It should be mentioned that limitations derived from Modern Portfolio Theory (Francis and Kim 2013) regarding the generation of the portfolios are out of the scope of this research as they do not affect its findings.

Finally, both the VaR_p baseline models and the proposed approach were evaluated for the period from 2018-01-01 to 2020-12-18, applying an 1-day rolling window forecast. To this end, the previous 125 days were used to calculate the correlation between the VaR_i's, interpolating the latter to a VaR_p.

3.2.3 Evaluation metrics

This section describes the evaluation metrics utilized to assess the proposed DeepVaR approach. These metrics are key towards interpreting the outcomes of the backtesting, as each one highlights different aspects of the results.

To begin with, a common variable in most of evaluation metrics is the *Hit Variable* associated with the ex-post observation of a VaR exception at time t is denoted as I_t where

$$I_t = \begin{cases} 1, & \text{if } \mathrm{VaR}_t > \mathrm{PnL}_t \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

Expected violations: are the maximum number of allowed exceedances (i.e. VaR > PnL). It is defined by the confidence probability α of VaR and the number of the days in the examined period.

The Expected Violations ($E[v]$) of VaR_α are

$$E[v] = (1-\alpha)N_{\text{days}}. \tag{8}$$

Number of violations: the number of the VaR violations is denoted as $N_{\text{violations}}$.

$$N_{\text{violations}} = \sum_{t=1}^{T} I_t. \tag{9}$$

Violation rate: the ratio of the Number of Violations over the examined period. This number should be less than the $1-\alpha$ to have a VaR^a model with good coverage.

$$r_{\text{Violations}} = \frac{N_{\text{Violations}}}{N_{\text{days}}}. \tag{10}$$

Quadratic Loss (l_{QL}): takes into account the quadratic magnitude of the exceedances

$$l_{\text{QL}} = \sum_{t=1}^{N} I_t(1 + (\mathrm{PnL}_t - \mathrm{VaR}_t)^2). \tag{11}$$

Smooth loss (l_Q) (González-Rivera et al. 2004): is a loss function that penalizes more heavily with weight (1–a) the observations for which PnL − VaR < 0. Smaller l_Q indicates a better goodness of fit

$$l_Q = \frac{1}{N}\sum_{t=1}^{N}(\alpha - (1+e^{dm})^{-1})m, \tag{12}$$

where $d = 25$, $m = \mathrm{PnL}_t - \mathrm{VaR}_t^\alpha$.

Tick loss (l_T): this loss penalizes exceedances with weight α and non-exceedances with weight $1-\alpha$, meaning that more conservative VaR estimations are producing higher Tick Loss

$$l_T = \sum_{t=1}^{N} (\alpha - I_t)(\mathrm{PnL}_t - \mathrm{VaR}_t^{\alpha}). \tag{13}$$

Firm loss (l_F) (Sarma et al. 2003): VaR is generally used by firms for internal risk management. However, there is a trade-off between the risk minimization and the profit maximization. A VaR estimator which reported 'too high' values of VaR would force the firm to hold 'too much' capital, imposing the opportunity cost of capital upon the firm

$$l_F = \begin{cases} (\mathrm{PnL}_t - \mathrm{VaR}_t)^2, & \text{if } \mathrm{PnL}_t < \mathrm{VaR}_t \\ -a\mathrm{VaR}_t, & \text{otherwise,} \end{cases} \tag{14}$$

where a measures the opportunity cost of capital and it was set equal to 1.

Additionally, according to Christoffersen (1998), VaR forecasts are valid if and only if the violation process I_t satisfies the following two assumptions: (i) The unconditional coverage (UC) hypothesis: the unconditional probability of a violation must be equal to the α coverage rate. (ii) The independence (IND) hypothesis: VaR violations observed at two different dates must be independently distributed

In the frame of our research, validity of these assumptions is tested by exploiting both the Christoffersen conditional coverage test and the Dynamic Quantile (DQ) test proposed by Engle and Manganelli (2004). The former jointly examines whether the percentage of exceptions is statistically equal to the one expected and whether VaR violations are serially independent. This is achieved by an independence test, which aims to reject VaR models with clustered violations. The likelihood ratio statistic of the conditional coverage test is $\mathrm{LR}_{\mathrm{cc}} = \mathrm{LR}_{\mathrm{uc}} + \mathrm{LR}_{\mathrm{ind}}$, which is asymptotically distributed X^2, and the $\mathrm{LR}_{\mathrm{ind}}$ statistic is the likelihood ratio statistic for the hypothesis of serial independence against the first-order Markov dependence. The latter examines whether the exception indicator I_t is uncorrelated with any variable that belongs to the information set Ω_{t-1} available when the VaR was calculated.

The main argument (hypothesis) of this work is that an RNN-based model is able to predict the future returns of the input time-series more accurately than well-established econometric models. The second argument lies in the fact that the utilized DeepAR model is fed with all the input time-series simultaneously, enabling cross-learning between them. As a result, changes in the dynamics of the one time-series may affect the predicted distributions of the other time-series.

Table 1 Mean running time to estimate VaR quantiles

Model	1 Asset (s)	4 Assets (s)	Rel. Difference (%)
DeepAR	12.457834	12.533903	0.61
HS	0.000201	0.000398	98.01
RM	0.003418	0.013461	293.83
GARCH	0.009435	0.038732	310.51
BiGAN	20.467264	20.947305	2.35
MC	0.000567	0.002108	271.78

Table 2 Performance of VaR$^{99\%}$ models in AUDUSD series

Model	$E[v]$	v	r_v	l_{QL}	l_Q	l_T	l_F
DeepVaR	9.28	**5**	**0.00539**	**0.00539**	**−0.00632**	**0.00020**	**0.02334**
HS	9.28	15	0.01616	0.01617	−0.00519	0.00022	0.02923
RM	9.28	21	0.02263	0.02263	−0.00472	0.00021	0.03457
GARCH	9.28	17	0.01832	0.01832	−0.00488	**0.00020**	0.03059
BiGAN	9.28	20	0.02155	0.02155	−0.00502	0.00023	0.03403
MC	9.28	18	0.01940	0.01940	−0.00489	0.00022	0.03149

Values in bold indicate the model(s) with the best performance per evaluation metric (column)

Table 3 Coverage and independence tests of VaR$^{99\%}$ models in AUDUSD series

Model	LR$_{uc}$	LR$_{ind}$	LR$_{cc}$	DQ
DeepVaR	2.386 [0.122]	0.054 [0.816]	2.441 [0.295]	2.158 [0.905]
HS	3.014 [0.083]	10.682** [0.001]	13.696** [0.001]	124.125** [0.0]
RM	11.036** [0.001]	2.931 [0.087]	13.966** [0.001]	31.907** [0.0]
GARCH	5.224* [0.022]	1.013 [0.314]	6.237* [0.044]	17.616** [0.007]
BiGAN	9.424** [0.002]	7.211** [0.007]	16.635** [0.0]	106.141** [0.0]
MC	6.512* [0.011]	8.445** [0.004]	14.957** [0.001]	108.176** [0.0]

The *p* values are in brackets

Table 4 Performance of VaR$^{99\%}$ models in GBPUSD series

Model	$E[v]$	v	r_v	l_{QL}	l_Q	l_T	l_F
DeepVaR	9.28	**7**	**0.00754**	**0.00754**	**−0.00617**	**0.00019**	0.02454
HS	9.28	8	0.00862	0.00862	−0.00557	0.00020	**0.02280**
RM	9.28	22	0.02371	0.02371	−0.00467	0.00021	0.03530
GARCH	9.28	14	0.01509	0.01509	−0.00485	**0.00019**	0.02710
BiGAN	9.28	16	0.01724	0.01724	−0.00502	0.00022	0.02969
MC	9.28	13	0.01401	0.01401	−0.00508	0.00021	0.02663

Values in bold indicate the model(s) with the best performance per evaluation metric (column)

Table 5 Coverage and independence tests of VaR$^{99\%}$ models in GBPUSD series

Model	LR$_{uc}$	LR$_{ind}$	LR$_{cc}$	DQ
DeepVaR	0.613 [0.434]	4.258* [0.039]	4.871 [0.088]	27.815** [0.0]
HS	0.184 [0.668]	3.713 [0.054]	3.897 [0.142]	31.297** [0.0]
RM	12.745** [0.0]	0.366 [0.545]	13.11** [0.001]	36.725** [0.0]
GARCH	2.108 [0.147]	1.623 [0.203]	3.731 [0.155]	35.0** [0.0]
BiGAN	4.055* [0.044]	4.868* [0.027]	8.923* [0.012]	98.703** [0.0]
MC	1.347 [0.246]	6.491* [0.011]	7.838* [0.02]	77.377** [0.0]

The *p* values are in brackets

Table 6 Performance of VaR$^{99\%}$ models in USDJPY series

Model	$E[v]$	v	r_v	l_{QL}	l_Q	l_T	l_F
DeepVaR	9.28	**8**	**0.00862**	**0.00862**	**− 0.00493**	0.00015	**0.02146**
HS	9.28	11	0.01185	0.01185	−0.00489	0.00016	0.02387
RM	9.28	24	0.02586	0.02586	−0.00366	0.00015	0.03456
GARCH	9.28	13	0.01401	0.01401	−0.00396	**0.00013**	0.02351
BiGAN	9.28	12	0.01293	0.01293	−0.00439	0.00016	0.02354
MC	9.28	12	0.01293	0.01293	−0.00460	0.00016	0.02411

Values in bold indicate the model(s) with the best performance per evaluation metric (column)

Table 7 Coverage and independence tests of VaR$^{99\%}$ models in USDJPY series

Model	LR$_{\text{uc}}$	LR$_{\text{ind}}$	LR$_{\text{cc}}$	DQ
DeepVaR	0.184 [0.668]	0.139 [0.709]	0.324 [0.851]	2.857 [0.827]
HS	0.308 [0.579]	2.477 [0.116]	2.785 [0.248]	49.276** [0.0]
RM	16.439** [0.0]	0.207 [0.649]	16.646** [0.0]	36.755** [0.0]
GARCH	1.347 [0.246]	0.37 [0.543]	1.717 [0.424]	8.448 [0.207]
BiGAN	0.743 [0.389]	7.143** [0.008]	7.886* [0.019]	81.484** [0.0]
MC	0.743 [0.389]	2.159 [0.142]	2.902 [0.234]	45.667** [0.0]

The *p* values are in brackets

Table 8 Performance of VaR$^{99\%}$ models in EURUSD series

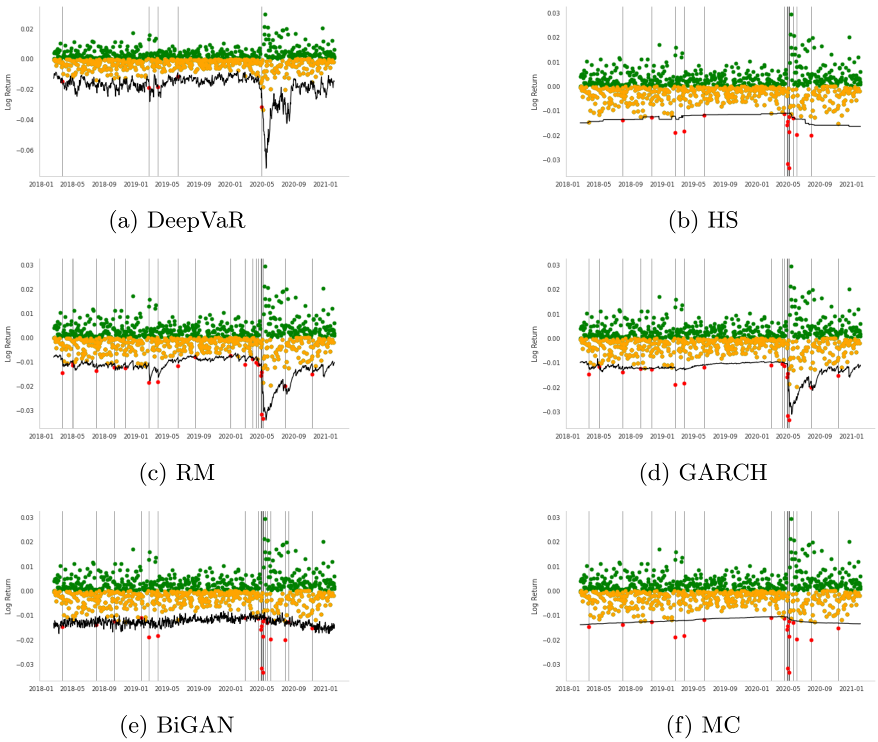
Model	$E[v]$	v	r_v	l_{QL}	l_Q	l_T	l_F
DeepVaR	9.28	**2**	**0.00216**	**0.00216**	**− 0.00508**	0.00014	0.01501
HS	9.28	4	0.00431	0.00431	−0.00459	0.00014	0.01539
RM	9.28	14	0.01509	0.01509	−0.00362	**0.00013**	0.02354
GARCH	9.28	5	0.00539	0.00539	−0.00404	**0.00013**	**0.01494**
BiGAN	9.28	8	0.00862	0.00862	−0.00404	0.00014	0.01814
MC	9.28	7	0.00754	0.00754	−0.00416	0.00014	0.01738

Values in bold indicate the model(s) with the best performance per evaluation metric (column)

Table 9 Coverage and independence tests of VaR$^{99\%}$ models in EURUSD series

Model	LR$_{\text{uc}}$	LR$_{\text{ind}}$	LR$_{\text{cc}}$	DQ
DeepVaR	8.463** [0.004]	0.009 [0.926]	8.472* [0.014]	5.786 [0.448]
HS	3.846 [0.05]	0.035 [0.852]	3.881 [0.144]	29.349** [0.0]
RM	2.108 [0.147]	0.429 [0.512]	2.537 [0.281]	9.598 [0.143]
GARCH	2.386 [0.122]	0.054 [0.816]	2.441 [0.295]	22.326** [0.001]
BiGAN	0.184 [0.668]	0.139 [0.709]	0.324 [0.851]	26.285** [0.0]
MC	0.613 [0.434]	0.107 [0.744]	0.72 [0.698]	28.095** [0.0]

The *p* values are in brackets



(a) DeepVaR

(b) HS

(c) RM

(d) GARCH

(e) BiGAN

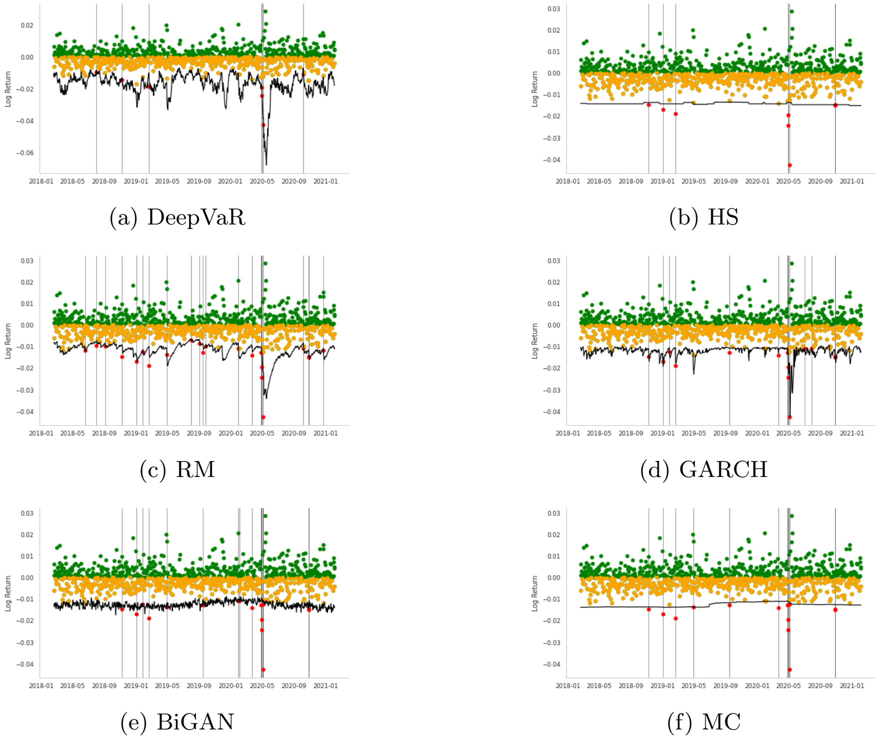
(f) MC

Fig. 3 AUDUSD: $VaR^{99\%}$ performance per model. In each figure, the VaR estimation of each model (black line) is depicted against the true PnL (green and yellow dots, for positive and negative returns, respectively). The red dots represent the VaR violations

The evaluation of the examined models in the following section proves that: (i) DeepVaR is able to capture (i.e., predict) the abrupt changes of the input time-series, and (ii) DeepVaR is the only model that avoids clustered VaR violations due to its non-linear nature, which also allows the proposed model to report uncorrelated VaR estimations.

3.2.4 Experiment details

GluonTS Alexandrov et al. (2019), Tensorflow Abadi et al. (2015), and arch Sheppard (2020) python libraries were utilized for the development of DeepVaR, BiGAN, and GARCH models, respectively, with NumPy (Harris et al. 2020) opted for the rest baseline models. Experiments are run on a desktop computer with an AMD Ryzen 5 5600x 6-Core CPU, 32GiB of RAM, and an NVIDIA GeForce RTX 3070 GPU. However, neural network training is performed in the CPU, and thus, further improvement in DeepVaR training time is feasible. Table 1 summarizes the required mean time in seconds per model to obtain the quantiles (e.g., q_1, q_{99}) needed for the VaR estimation of both single and four assets portfolios. The fourth column of



(a) DeepVaR

(b) HS

(c) RM

(d) GARCH

(e) BiGAN

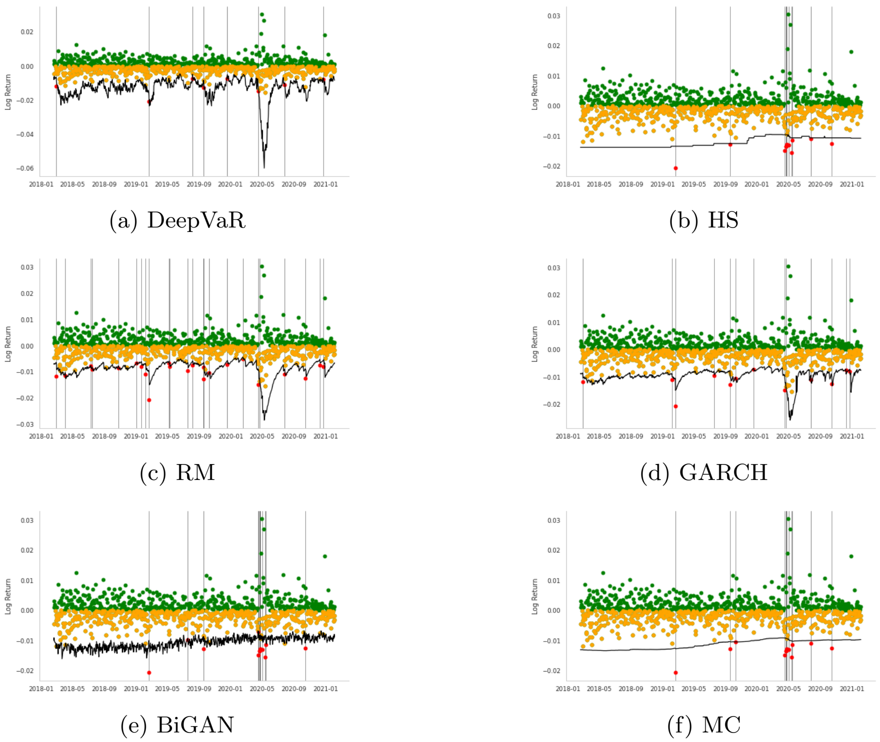
(f) MC

Fig. 4 GBPUSD: VaR$^{99\%}$ performance per model. In each figure, the VaR estimation of each model (black line) is depicted against the true PnL (green and yellow dots, for positive and negative returns, respectively). The red dots represent the VaR violations

the table indicates the relative difference in calculation time between the two different input sizes. According to these findings, it is obvious that despite the fact that deep learning-based models require significantly more time to estimate VaR than the other models, that time ($\approx 12.5s$) is very low, enabling VaR estimation even for intra-day trading applications. Moreover, the input size has a minimal effect on deep learning models' training time which leverage matrices operations to parallelize computations. In contrast, estimation time in econometric models such as GARCH is linearly dependent on the number of the input time-series.

4 Univariate VaR performance

In this section, initially, we present the VaR estimations for each FX asset separately for the period 2018-01-01 to 2020-12-18, following the dataset descriptions Sect. 3.2.2. Tables 2, 3, 4, 5, 6, 7, 8, 9 and Figs. 3, 4, 5, 6 summarize the performance of each model. In each figure, the VaR estimation of each model (black line) is depicted against the true portfolio returns (green and yellow dots, for positive



(a) DeepVaR

(b) HS

(c) RM

(d) GARCH

(e) BiGAN

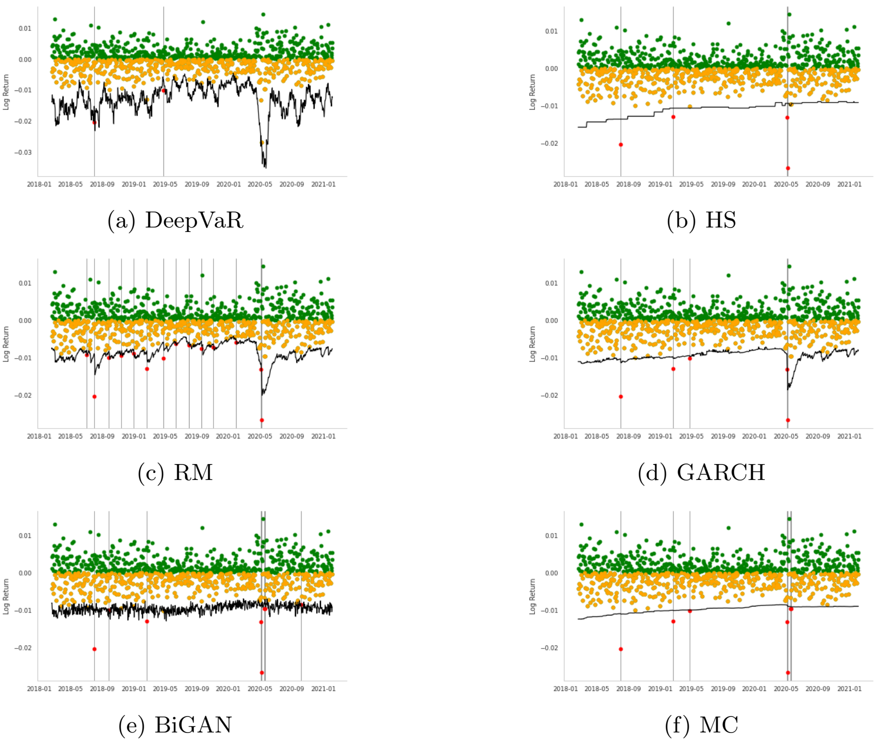
(f) MC

Fig. 5 USDJPY: VaR$^{99\%}$ performance per model. In each figure, the VaR estimation of each model (black line) is depicted against the true PnL (green and yellow dots, for positive and negative returns, respectively). The red dots represent the VaR violations

and negative returns, respectively). The red dots are the VaR violations. The complementary tables following each model's presentation contain the aforementioned evaluation metrics and statistical tests.

The first time-series under consideration is the AUDUSD currency pair which is characterized by very strong liquidity due to the large amount of Australian exports. As depicted in Table 2, the DeepVaR model outperforms the rest of the models in this time-series, having the lowest loss over all the examined loss metrics. Additionally, according to Table 3, which presents the results of Christoffersen and DQ tests for 99% VaR estimation, all the models, except DeepVaR, have been rejected for not having the "correct unconditional coverage". These outcomes can also be seen in Fig. 3. The advantage of the DeepVaR model emerges from the fact that it adapts to stricter VaR estimates as AUDUSD volatility increases, while classic VaR models suffer from clustered VaR violations (e.g., at the end of May 2021).

The next examined FX instrument is GBPUSD associated with two of the largest western economies with very strong trading relationships. According to Table 4, the DeepVaR model has the lowest number of violations, quadratic, smooth, and tick loss, while the GARCH model shares similar performance to DeepVaR in terms of



(a) DeepVaR

(b) HS

(c) RM

(d) GARCH

(e) BiGAN

(f) MC

Fig. 6 EURUSD: VaR$^{99\%}$ performance per model. In each figure, the VaR estimation of each model (black line) is depicted against the true PnL (green and yellow dots, for positive and negative returns respectively). The red dots represent the VaR violations

tick loss, with the HS model illustrating the lowest firm loss. Figure 4 indicates that DeepVaR and HS fail to capture the negative returns almost at the same dates, while the other two models have more frequent VaR violations. Results in Table 5 denote that all the models, except from the RiskMetrics model, passed the Christoffersen's test; however, all the examined models failed in the DQ test. Thus, the VaR violations of each model in GBPUSD time-series are serially dependent, although Fig. 4 shows that this issue is more pronounced in the RM and GARCH models.

VaR is also estimated for the US Dollar to Japanese Yen currency pair (USDJPY) which is the second most commonly traded pair after EURUSD. Generally, USDJPY has very high liquidity; however, JPY can also be viewed as a 'safe haven' currency during periods of global economic uncertainty. As shown in Table 6, DeepVaR model has the best overall performance, besides tick loss where the GARCH model performs better. As far as the coverage and independence tests illustrated in Table 7, both the DeepVaR and the GARCH models showcase promising results. However, only DeepVaR has less VaR violations than the nominal threshold. It is also noted that DeepVaR reports stricter VaR estimates than the rest of the models during the high volatility period in May

2020, as shown in Fig. 5. This is due to the shared dependencies between the four examined time-series during the training of the DeepAR algorithm.

The last utilized currency pair, EURUSD, is the most widely traded forex pair in the market as it comprises the currencies of two of the world's biggest economies. The results of the various loss metrics are summarized in Table 8. DeepVaR presents the lowest quadratic and smooth loss, as well as the fewest VaR violations. GARCH model performed better than the other models in terms of tick and firm loss. In addition, Table 9 shows that all models passed Christoffersen coverage and independence test, but only DeepVaR and RM succeeded in the DQ test. Finally, Fig. 6 shows that DeepVaR is the only model that did not suffer any VaR violation during the EURUSD volatility shift in May 2020.

All the aforementioned results, for each of the FX time-series, could be explained by the individual characteristics of the examined models, with the adaptation of each model to the returns' volatility being the most influential. The VaR models that fit closer to the true returns would potentially suffer from VaR breaches, while the ones that generalize better would produce a greater firm loss (Sect. 3.2.3).

More specifically, as stated in Sect. 3.2.1, the HS model uses the last 1000 historical returns of the input time-series to estimate VaR. As a result, in order for the HS to alter to a stricter estimation, the most recent daily returns would have to be lower than the worst 10 (for 99% confidence interval) of the 1000-day historical window. Using such a large history, HS is able to capture most of the negative returns. However, in case of a sudden and permanent change in volatility, the adaptation of this model to the new inputs would be slow. This is also evident in Figs. 3, 4, 5 and 6 with HS being a straight line for long periods of time. Similar are the findings for the MC model as its input parameters (i.e., μ, σ^2) have been derived from the last 900 historical data.

As for the BiGAN model, Figs. 3, 4, 5 and 6 illustrate that the estimated VaR oscillates heavily between a short range of values as the generated distributions slightly variate between each day of the testing period. However, BiGAN fails to capture the sudden negative returns, indicating that this model could not efficiently predict VaR in the case of rare financial events.

On the other hand, the other two baseline models are both GARCH(1,1) type which are capable of capturing the time-varying volatility of returns. Although, their main difference is the historical data used to estimate their parameters. The effective historical data for the RM is set to 74 days, while for the GARCH model is 900 (Sect. 3.2.1) containing information from a quite larger information set. This explains the fact that the RM prediction is closer to the true PnL than the GARCH and HS models, producing the most VaR violations, for all the examined time-series, among the utilized models.

This trade-off between VaR model validity and adaptation to time-series variance is mitigated by DeepVaR. The latter, being trained for a period of 900 days, predicts the parameters of the returns distribution based on the last 15 values (context length) of the input series. In such way, DeepVaR is able to both "memorize" past information during training and optimally capture the time-series volatility using recent data for parameters' estimation.

5 Multivariate VaR performance

Apart from the univariate evaluation of each VaR estimation method, the performance of each model was evaluated in a realistic multivariate perspective, hence in the context of 1000 random portfolios. These portfolios were created randomly by producing both positive and negative positions on the aforementioned FX assets, with the absolute sum of the positions(weights) to be equal to one. It should be mentioned that these portfolios have not any "hold" position, which means that every time all available capital is allocated. This approach is not far from the real-life portfolio management as it is a common strategy towards minimizing commission fees. In addition, no commission fees or extra charges were taken into consideration to simplify the evaluation schema, given that they could be modeled as a constant common term in every VaR model without affecting the results drastically. On the contrary, to compute the VaR of a portfolio, the correlation ρ among the FX instruments should be taken into account (Longerstaey and Spencer 1996). In this case, the VaR of a portfolio for a given day can be estimated by Eq. 15

$$\mathrm{VaR}_p^{\alpha} = \sqrt{\mathrm{VRV}^T}, \tag{15}$$

where V is a vector of the weighted VaR estimates per instrument $V = [w_1\mathrm{VaR}_1^{\alpha}, w_2\mathrm{VaR}_2^{\alpha}, w_3\mathrm{VaR}_3^{\alpha}, w_4\mathrm{VaR}_4^{\alpha}]$ and R is the correlation matrix of FX assets' daily returns, with the last 125 daily returns of the assets to be taken into account for the calculation of R matrix

$$R = \begin{pmatrix} 1 & \rho_{1,2} & \rho_{1,2} & \rho_{1,4} \\ \rho_{2,1} & 1 & \rho_{2,3} & \rho_{2,4} \\ \rho_{3,1} & \rho_{3,2} & 1 & \rho_{3,4} \\ \rho_{4,1} & \rho_{4,2} & \rho_{4,3} & 1 \end{pmatrix}.$$

Furthermore, for long positions ($w_i > 0$) VaR_i^{α} is used, while for short positions ($w_i < 0$) $\mathrm{VaR}_i^{1-\alpha}$, where α the confidence probability of VaR estimation and i is the corresponding FX asset. The procedure of portfolio VaR estimation is summarized in Algorithm 2.

Table 10 Average performance of $\mathrm{VaR}^{99\%}$ models over the FX portfolios

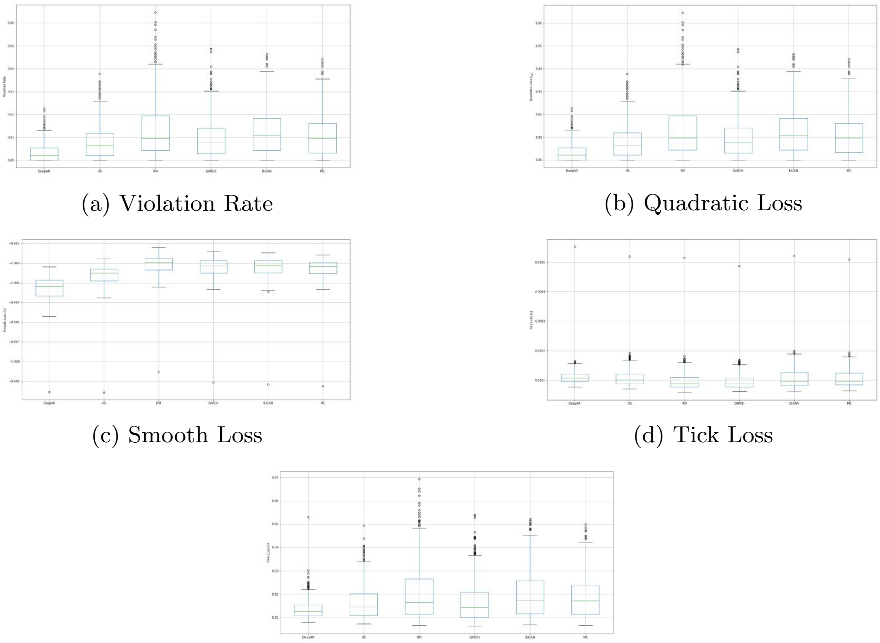
Model	$E[v]$	v	r_v	l_{QL}	l_Q	l_T	l_F
DeepVaR	9.28	**2.90310**	**0.00319**	**0.00313**	**− 0.00427**	0.00011	**0.01351**
HS	9.28	7.29	0.00784	0.00784	−0.00361	0.00011	0.01618
RM	9.28	12.14	0.01308	0.01308	−0.00305	**0.00010**	0.02002
GARCH	9.28	8.61	0.00928	0.00928	−0.00321	**0.00010**	0.01660
BiGAN	9.28	11.51	0.01240	0.01240	−0.00320	0.00011	0.01966
MC	9.28	10.31	0.01111	0.01111	−0.00327	0.00011	0.01854

Values in bold indicate the model(s) with the best performance per evaluation metric (column)

Table 11 Percentage of portfolios passed the coverage and independence tests of VaR$^{99\%}$ per model in significant level 95%

Model	LR_{uc}	LR_{ind}	LR_{cc}	DQ
DeepVaR	72.8	**95.5**	**80.6**	**84.6**
HS	**76.9**	72.0	59.5	36.7
RM	65.2	95.3	68.3	55.9
GARCH	76.5	92.9	77.5	63.0
BiGAN	64.3	71.5	53.5	26.1
MC	70.4	68	54.8	26.7

Values in bold indicate the model(s) with the best performance per evaluation metric (column)



(a) Violation Rate

(b) Quadratic Loss

(c) Smooth Loss

(d) Tick Loss

(e) Firm Loss

Fig. 7 Box-plots of the $VaR^{99\%}$ performance per model over 1000 random portfolios. Each sub-figure refers to a different loss function. **a** and **b** Show that DeepVaR is the only model having violation rate and quadratic loss lower than 1-α (i.e., 0.01) confidence probability over most of the random portfolios. In terms of Smooth Loss (c), the superiority of DeepVaR over the rest of the models is evident. Tick loss (d) is the only metric where DeepVaR under-performs compared to the rest of the models. e Presents the results of VaR estimation firm loss, with DeepVaR to be the winning model

Algorithm 2 Portfolio VaR rolling window estimation

```
Generate random portfolio weights $w \in \mathbb{R}^{1\times 4}$
Set confidence probability $a$ (i.e. $\alpha = 0.99$) of $VaR$ estimation
Set window $t_w = 125$, for the calculation of returns correlation matrix $R \in \mathbb{R}^{4\times 4}$
Split FX returns to train $R_{train} \in \mathbb{R}^{125\times 4}$ and test set $R_{test} \in \mathbb{R}^{928\times 4}$
for test day $t = 1$ to $T$ do
    $R \Leftarrow \textsc{corr}(R_{train})$
    for $m$ in $models$ do
        Initialize zero vector $V \in \mathbb{R}^{1\times 4}$
        for $i$ in $w$ do
            if $w_i < 0$ then
                $V_i \Leftarrow w_i VaR^{1-\alpha}_{i,m,t}$
            else
                $V_i \Leftarrow w_i VaR^{\alpha}_{i,m,t}$
            end if
        end for
        $VaR^{\alpha}_{m,t} \Leftarrow -\sqrt{VRV^T}$
    end for
    Append $R_{test}[t]$ to $R_{train}$
    $PnL_t \Leftarrow \sum_{i=1}^{4} w_i R_{train,i}[-1]$
end for
```

Finally, the average performance of each model over the random portfolios is summarized in Table 10. As presented in this table, DeepVaR achieved by far the lowest loss in all loss functions included, besides tick loss. Table 11 illustrates the percentage of the random portfolios per model passed the coverage and independence tests, depicting that DeepVaR is by far the most valid model.

Moreover, the loss metrics are depicted in box-plots (Fig. 7) to provide a better overview of the VaR models' performance over several random portfolios. Specifically, Fig. 7a shows that DeepVaR is the only model having a violation rate lower than $1 - \alpha$ (i.e., 0.01) confidence probability over most of the portfolios. The violation rate of the other three models examined is highly dependent on the portfolio composition, while in some cases, the violation rate is significantly higher than the nominal threshold. Similar findings are derived from Quadratic Loss function (Fig. 7b).

In terms of Smooth Loss, which penalizes more heavily VaR violations, the superiority of DeepVaR over the rest of the models is evident (Fig. 7c). As far as the Tick Loss is concerned, this is the only metric where DeepVaR under-performs compared to the other models. However, DeepVaR has stable tick loss regardless of the portfolio, while in the other models, this metric has higher volatility among the different portfolios.

The last examined loss function, Firm Loss, takes into account the opportunity cost of capital, where firms would unnecessarily reserve capital according to the

VaR estimates of their portfolios. Figure 7e presents the results of VaR estimation firm loss for all the portfolios, with DeepVaR to be again the winning model.

6 Conclusions

This paper aims at addressing one of the main challenges in the financial sector, which is the constant search and development of more accurate risk estimation models. The utilization of Deep Neural Networks tools and techniques can introduce an innovative approach for VaR estimation, aiming towards a robust risk management framework.

To this direction, different parametric semi-parametric and non-parametric VaR estimation approaches were incorporated as baseline models, while an innovative framework based on a probabilistic Deep Neural Network was analyzed and compared against the baseline ones. The proposed framework yields better results in terms of the utilized evaluation metrics. More specifically, it is more effective than the others in terms of VaR violations and excess loss beyond the VaR threshold, while at the same time, it permits financial institutions to reserve less capital on liquid assets compared to the classical approaches. Additionally, the framework has the capability to incorporate other VaR models through the framework's modularity.

Future work will focus on improving the efficiency of the proposed probabilistic approach in terms of high-frequency trading. To achieve the latter, intra-day data should be leveraged along with parallel and distributed computing techniques. It should also be highlighted that the hyper-parameter tuning of any deep neural network is a highly time-consuming task requiring extensive computational resources (Diaz et al. 2017). As a result, only a specific range of them, based on randomized grid search of the hyper-parameter space, was tested and evaluated in the context of our research, while further improvement of the model performance may be achieved through further hyper-parameters tuning. Finally, additional sources of complementary information could be integrated for improved results, such as sentiment analysis on texts (i.e., tweets and financial news) which may have an impact on the market movements.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s42521-022-00050-0.

Acknowledgements The research leading to the results presented in this paper has received funding from the European Union's funded Project INFINITECH under Grant Agreement No. 856632.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

Abad, P., Benito, S., & López, C. (2014). A comprehensive review of value at risk methodologies. *The Spanish Review of Financial Economics, 12*(1), 15–32.

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., et al. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Retrieved from https://www.tensorflow.org/ (Software available from tensorflow.org)

Alexandrov, A., Benidis, K., Bohlke-Schneider, M., Flunkert, V., Gasthaus, J., Januschowski, T., et al. (2019). Gluonts: Probabilistic time series models in python. arXiv preprint arXiv:1906.05264.

Angelidis, T., & Degiannakis, S.A. (2018). Backtesting var models: A two-stage procedure. Available at SSRN 3259849.

Bekiros, S. D., & Georgoutsos, D. A. (2005). Estimation of value-at-risk by extreme value and conventional methods: A comparative evaluation of their predictive performance. *Journal of International Financial Markets, Institutions and Money, 15*(3), 209–228.

Bollerslev, T., Chou, R. Y., & Kroner, K. F. (1992). Arch modeling in finance: A review of the theory and empirical evidence. *Journal of Econometrics, 52*(1–2), 5–59.

Chang, Y.-P., Hung, M.-C., & Wu, Y.-F. (2003). Nonparametric estimation for risk in value-at-risk estimator. *Communications in Statistics-Simulation and Computation, 32*(4), 1041–1064.

Christoffersen, P.F. (1998). Evaluating interval forecasts. *International Economic Review*, *39*(4), 841–862.

Christoffersen, P., Hahn, J., & Inoue, A. (2001). Testing and comparing value-at-risk measures. *Journal of Empirical Finance, 8*(3), 325–342.

Das. (2021). *World economic outlook: Managing divergent recoveries*. International Monetary Fund.

Dattels, P., & Miyajima, K. (2009). Will emerging markets remain resilient to global stress? *Global Journal of Emerging Market Economies, 1*(1), 5–24.

De Waal, B., Petersen, M. A., Hlatshwayo, L. N., & Mukuddem-Petersen, J. (2013). A note on basel iii and liquidity. *Applied Economics Letters, 20*(8), 777–780.

Diaz, G. I., Fokoue-Nkoutche, A., Nannicini, G., & Samulowitz, H. (2017). An effective algorithm for hyperparameter optimization of neural networks. *IBM Journal of Research and Development, 61*(4/5), 9–1.

Donahue, J., Krähenbühl, P., Darrell, T. (2016). Adversarial feature learning. arXiv preprint arXiv:1605.09782 .

Einhorn, D., & Brown, A. (2008). Private profits and socialized risk. *Global Association of Risk Professionals, 42,* 10–26.

Elsinger, H., Lehar, A., & Summer, M. (2006). Risk assessment for banking systems. *Management Science, 52*(9), 1301–1314.

Engle, R. F., & Manganelli, S. (2004). Caviar: Conditional autoregressive value at risk by regression quantiles. *Journal of Business & Economic Statistics, 22*(4), 367–381.

Francis, J. C., & Kim, D. (2013). *Modern portfolio theory: Foundations, analysis, and new developments* (Vol. 795). John Wiley & Sons.

Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual prediction with lstm. *Neural Computation, 12*(10), 2451–2471.

Gneiting, T., Raftery, A., Balabdaoui, F., Westveld, A. (2004). Verifying prob-abilistic forecasts: Calibration and sharpness. Preprints, 17th conf. on probability and statistics in the atmospheric sciences, seattle, wa, amer. meteor. soc (Vol. 2).

Gneiting, T., Raftery, A. E., Westveld, A. H., III., & Goldman, T. (2005). Calibrated probabilistic forecasting using ensemble model output statistics and minimum CRPs estimation. *Monthly Weather Review, 133*(5), 1098–1118.

González-Rivera, G., Lee, T.-H., & Mishra, S. (2004). Forecasting volatility: A reality check based on option pricing, utility function, value-at-risk, and predictive likelihood. *International Journal of forecasting, 20*(4), 629–645.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). Generative adversarial nets. *Advances in Neural Information Processing Systems*, *27*, 2672–2680.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., et al. (2020). Array programming with numpy. *Nature, 585*(7825), 357–362.

Hendricks, D. (1996). Evaluation of value-at-risk models using historical data. *Economic Policy Review*, *2* (1), 39–69.

Khan, M. A., Saqib, S., Alyas, T., Rehman, A. U., Saeed, Y., Zeb, A., & Mohamed, E. M. (2020). Effective demand forecasting model using business intelligence empowered with machine learning. *IEEE Access, 8,* 116013–116023.

Kingma, D.P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Kuester, K., Mittnik, S., & Paolella, M. S. (2006). Value-at-risk prediction: A comparison of alternative strategies. *Journal of Financial Econometrics, 4*(1), 53–89.

Lim, B., & Zohren, S. (2020). Time series forecasting with deep learning: A survey. arXiv preprint arXiv:2004.13408.

Longerstaey, J., & Spencer, M. (1996). Riskmetricstm-technical document. *Morgan Guaranty Trust Company of New York: New York, 51,* 54.

Makridis, G., Mavrepis, P., Kyriazis, D., Polychronou, I., Kaloudis, S. (2020). Enhanced food safety through deep learning for food recalls prediction. *International conference on discovery science* (pp. 566–580).

Mcneil, A.J. (1998). Calculating quantile risk measures for financial return series using extreme value theory. ETHZurich. https://doi.org/10.3929/ethz-a-004320029, https://www.research-collection.ethz.ch/handle/20.500.11850/146132.

Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2021). A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR), 54*(6), 1–35.

Mohebali, B., Tahmassebi, A., Meyer-Baese, A., Gandomi, A.H. (2020). Prob-abilistic neural networks: A brief overview of theory, implementation, and application. *Handbook of Probabilistic Models*, pp. 347–367.

Montavon, G., Orr, G., & Müller, K.-R. (2012). *Neural networks: Tricks of the trade* (Vol. 7700). Springer.

Neuneier, R. (1996). Optimal asset allocation using adaptive dynamic programming. *Advances in Neural Information Processing Systems*, *8*, 952–958.

Novak, S. Y. (2011). *Extreme value methods with applications to finance*. CRC Press.

Pfenninger, M., Rikli, S., Bigler, D.N. (2021). Wasserstein gan: Deep generation applied on financial time series. Available at SSRN 3877960.

Salinas, D., Flunkert, V., Gasthaus, J., & Januschowski, T. (2020). Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting, 36*(3), 1181–1191.

Sarma, M., Thomas, S., & Shah, A. (2003). Selection of value-at-risk models. *Journal of Forecasting, 22*(4), 337–358.

Sen, R., Yu, H.-F., Dhillon, I. (2019). Think globally, act locally: A deep neural network approach to high-dimensional time series forecasting. arXiv preprint arXiv:1905.03806.

Sheppard, K. (2020). bashtage/arch: Release 4.15 (version 4.15). Zenodo. https://doi.org/10.5281/zenodo, 593254 .

So, M. K., & Philip, L. (2006). Empirical analysis of Garch models in value at risk estimation. *Journal of International Financial Markets, Institutions and Money, 16*(2), 180–197.

Weng, B., Ahmed, M. A., & Megahed, F. M. (2017). Stock market one-day ahead movement prediction using disparate data sources. *Expert Systems with Applications, 79,* 153–163.

Wu, S., Xiao, X., Ding, Q., Zhao, P., Wei, Y., Huang, J. (2020). Adversarial sparse transformer for time series forecasting. H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems* (Vol. 33, pp. 17105–17115). Curran Associates.

Xiong, Z., Liu, X.-Y., Zhong, S., Yang, H., Walid, A. (2018). Practical deep reinforcement learning approach for stock trading. arXiv preprint arXiv:1811.07522.

Yamai, Y., Yoshiba, T., et al. (2002). Comparative analyses of expected shortfall and value-at-risk: Their estimation error, decomposition, and optimization. *Monetary and Economic Studies, 20*(1), 87–121.

Yan, H., & Ouyang, H. (2018). Financial time series prediction based on deep learning. *Wireless Personal Communications, 102*(2), 683–700.

Zhao, K. (2020). Analysis of investment risk assessment model of financial institutions under economic growth. *International conference on urban intelligence and applications* (pp. 233–240).

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Georgios Fatouros[1,2] · **Georgios Makridis**[1] · **Dimitrios Kotios**[1] · **John Soldatos**[2] · **Michael Filippakis**[1] · **Dimosthenis Kyriazis**[1]

Georgios Makridis
gmakridis@unipi.gr

Dimitrios Kotios
dimkotios@unipi.gr

John Soldatos
jsoldat@innov-acts.com

Michael Filippakis
mfilip@unipi.gr

Dimosthenis Kyriazis
dimos@unipi.gr

[1] Department of Digital Systems, University of Piraeus, Karaoli and Dimitriou 80, 18534 Piraeus, Greece

[2] Innov-Acts Ltd, Kolokotroni 6, 1101 Nicosia, Cyprus