

VOLGAN:

a generative model for arbitrage-free implied volatility surfaces

Milena Vuletić* and Rama Cont
Mathematical Institute, University of Oxford

Revision: February 2025

Abstract

We introduce VOLGAN, a generative model for arbitrage-free implied volatility surfaces. The model is trained on time series of implied volatility surfaces and underlying prices and is capable of generating realistic scenarios for joint dynamics of the implied volatility surface and the underlying asset. We illustrate the performance of the model by training it on SPX implied volatility time series and show that it is able to learn the covariance structure of the co-movements in implied volatilities and generate realistic dynamics for the (VIX) volatility index. In particular, the generative model is capable of simulating scenarios with non-Gaussian distributions of increments for state variables as well as time-varying correlations.

Finally, we illustrate the use of VOLGAN to construct data-driven hedging strategies for option portfolios, and show that these strategies can outperform Black-Scholes delta and delta-vega hedging.

JEL Classification: G13, G17, C15, C22, C45, C53, C63

Keywords: generative models, GenAI, scenario simulation, volatility surface, options, implied volatility.

*Corresponding author. Email: Milena.Vuletic@maths.ox.ac.uk

Contents

1	Introduction	3
2	Implied volatility surfaces: shape constraints and dynamics	4
2.1	Static arbitrage and shape constraints	5
2.2	Dynamics of implied volatility co-movements	6
3	A generative model for implied volatility surfaces	6
3.1	Architecture	7
3.2	Training objective	9
3.3	Scenario re-weighting	10
3.4	Numerical implementation	11
4	Learning to simulate SPX implied volatility surfaces	13
4.1	Data	13
4.2	Out-of-sample performance	14
4.2.1	Detecting extreme market events	14
4.2.2	Smoothness and arbitrage constraints	15
4.2.3	Next-day forecasting	18
4.2.4	Distributions and correlations learned by the generator .	25
4.2.5	Principal component analysis	27
4.2.6	Correlation structure of variables	31
5	Application to hedging and risk management of option portfolios	34
5.1	Example: hedging a straddle	35

1 Introduction

Option prices are quoted in terms of their *implied volatilities*, which are obtained by inverting the Black-Scholes formula given the market prices of options. The implied volatility surface, which summarises the cross-section of option prices across strikes and maturities, gives a snapshot of the state of the options market. The dependence of implied volatility on moneyness and time-to-maturity, which is referred to as the *smile*, *skew* and *term structure* have inspired the development of alternative option pricing models [Gatheral, 2011, Heston, 1993, Cont and Tankov, 2004]. Any such option pricing model implies a model for the cross-sectional dependence of implied volatilities on strike and maturity, as well as their dynamics across time. However, this dynamics is typically intractable and there has been an interest from practitioners in directly modelling the dynamics of implied volatility as a state variable [Schönbucher, 1999, Babbar, 2001, Cont and da Fonseca, 2002, Cont et al., 2002, Durrleman, 2010, Cont and Vuletic, 2023, Avellaneda et al., 2020]. Such ‘market models’ of implied volatility should appropriately capture the co-movements of implied volatilities across moneyness and time-to-maturity, reproduce the empirically observed dynamics of implied volatilities [Cont and da Fonseca, 2002], be able to capture the smile, skew, and term structure, and satisfy arbitrage constraints [Davis and Hobson, 2007, Gerhold and Gülmüm, 2020].

Given the high dimensionality of the volatility surface and the complexity of its dynamics, it is challenging to capture all these properties in a parametric model. It is therefore of interest to examine whether a data-driven approach can be used to overcome these modelling challenges.

Contribution In the present work we introduce VOLGAN, a *fully data-driven* generative model for the dynamic simulation of arbitrage-free implied volatility surfaces. Our model is trained on a time series of market-quoted implied volatilities and is capable of generating realistic dynamic scenarios for implied volatility surfaces. We illustrate the performance of the model by training it on SPX implied volatility time series and show that it is able to learn the covariance structure of co-movements in implied volatilities and generate realistic dynamics for the (VIX) volatility index [CBOE, 2022]. In particular, the generative model is capable of simulating scenarios with non-Gaussian distributions of increments for state variables as well as time-varying correlations.

Last but not least, we show that VOLGAN may be used to compute data-driven hedging strategies for option portfolios. Using examples of SPX option portfolios, we show that VOLGAN can produce hedge ratios with better performance than Black-Scholes delta hedging and delta-vega hedging, with automatic selection of the hedging instruments. In contrast with model-based approaches such as Deep hedging [Buehler et al., 2019], our approach is completely *data-driven* and model-free, in the spirit of the pioneering work of [Hutchinson et al., 1994].

Our model builds on previous work on generative adversarial networks (GANs) for scenario simulation in finance, starting with [Takahashi et al., 2019] and [Wiese et al., 2020] for price dynamics. More recently, GAN methods have

been deployed for scenario simulation in options markets. [Wiese et al., 2019] and [Wiese et al., 2021] use a classical GAN approach. [Cuchiero et al., 2020] and [Cohen et al., 2022] use a "neural SDE" to parameterize volatility surface dynamics. [Cao et al., 2020] use a supervised learning approach to extract information from historical implied volatility dynamics, while [Ning et al., 2023] combines SDEs with Variational Autoencoders [Kingma et al., 2019].

In contrast with the aforementioned approaches which deploy the classical GAN methodology of [Goodfellow et al., 2014] using binary cross-entropy (BCE) as a training objective, we propose a bespoke training criterion adapted to the financial application at hand, as advocated in [Cont et al., 2022] and [Vuletić et al., 2023], combined with a scenario weighting approach based on [Cont and Vuletic, 2023] to take care of arbitrage constraints.

Outline. Section 2 summarizes properties of implied volatility surfaces and outlines some desirable requirements for a dynamic model of implied volatility. Section 3 describes VOLGAN, our proposed generative model for implied volatility surfaces. Section 4 presents the results obtained by training VOLGAN on SPX implied volatility data and discusses the model's ability to produce realistic scenarios for implied volatility co-movements and the VIX index. Section 5 demonstrates applications of VOLGAN for hedging and shows that hedging strategies computed using VOLGAN can outperform commonly used delta hedging and delta-vega hedging strategies.

2 Implied volatility surfaces: shape constraints and dynamics

Denoting the price of the underlying asset by S_t , the implied volatility may be parameterized in terms of moneyness $m = K/S_t$ and time to maturity $\tau = T - t$ of the option. The implied volatility associated with a call option with moneyness m and time-to-maturity τ on a non-dividend paying asset S is the unique value $\sigma_t(m, \tau)$ such that the Black-Scholes price $C_{BS}(S_t, K, \tau, \sigma_t(m, \tau))$ matches the market price $C_t(m, \tau)$ of the call:

$$C_t(m, \tau) = C_{BS}(S_t, K, \tau, \sigma_t(m, \tau)) = S_t N(d_1) - K e^{-r\tau} N(d_2)$$

$$d_1 = \frac{-\ln m + \tau(r + \frac{\sigma^2}{2})}{\sigma \sqrt{\tau}} \quad d_2 = \frac{-\ln m + \tau(r - \frac{\sigma^2}{2})}{\sigma \sqrt{\tau}},$$

where N is the c.d.f of a standard Gaussian $\mathcal{N}(0, 1)$ variable. The implied volatility surface $\sigma_t(m, \tau)$ at date t provides a snapshot of options prices in the market [Gatheral, 2011]: specifying the implied volatility surface is equivalent to specifying the prices of all European calls and puts available in the market, given the current term structure of interest rates and dividends.

2.1 Static arbitrage and shape constraints

It has been empirically observed that implied volatilities of call and put options in listed options markets exhibit a dependence on exercise price K and maturity date T [Cont and da Fonseca, 2002, Dumas et al., 1998, Dupire, 1994, Gatheral, 2011] (or, alternatively, on the moneyness $m = K/S_t$ and time-to-maturity $\tau = T - t$). However not every cross-sectional profile for the function $(m, \tau) \mapsto \sigma_t(m, \tau)$ is admissible, as the resulting call/put option prices should satisfy certain *static arbitrage constraints* [Davis and Hobson, 2007, Gerhold and Gültüm, 2020]. In particular call option prices should be:

- increasing in time to maturity: $\partial_\tau C_{BS}(S_t, K, \tau, \sigma_t(m, \tau)) \geq 0$,
- decreasing in moneyness: $\partial_m C_{BS}(S_t, K, \tau, \sigma_t(m, \tau)) \leq 0$,
- convex in moneyness: $\partial_m^2 C_{BS}(S_t, K, \tau, \sigma_t(m, \tau)) \geq 0$.

These constraints translate to nonlinear inequalities involving σ_t , $\partial_m \sigma_t$, $\partial_m^2 \sigma_t$, $\partial_\tau \sigma_t$ [Cont et al., 2002], which in turn impose constraints on the possible shapes of the implied volatility surface $\sigma_t(m, \tau)$.

Given a fixed grid in moneyness and time to maturity

$$(\mathbf{m}, \boldsymbol{\tau}) = (m_i, \tau_j)_{i=1, \dots, N_m; j=1, \dots, N_\tau},$$

with $m_i < m_{i+1}$ and $\tau_j < \tau_{j+1}$, we define the relative call prices

$$c(m, \tau) := \frac{1}{S} C_{BS}(S, K, \tau, \sigma) = N(d_1) - m e^{-r\tau} N(d_2). \quad (1)$$

Following [Cont and Vuletic, 2023], we define the *arbitrage penalty* associated with the (discretely sampled) volatility surface $\boldsymbol{\sigma}(\mathbf{m}, \boldsymbol{\tau})$ as:

$$\Phi(\boldsymbol{\sigma}(\mathbf{m}, \boldsymbol{\tau})) = p_1(\boldsymbol{\sigma}(\mathbf{m}, \boldsymbol{\tau})) + p_2(\boldsymbol{\sigma}(\mathbf{m}, \boldsymbol{\tau})) + p_3(\boldsymbol{\sigma}(\mathbf{m}, \boldsymbol{\tau})). \quad (2)$$

where the functions p_1, p_2, p_3 measure violations of calendar, call and butterfly arbitrage constraints, respectively:

$$p_1(\boldsymbol{\sigma}(\mathbf{m}, \boldsymbol{\tau})) = \sum_{i=1}^{N_m} \sum_{j=1}^{N_\tau} \left(\tau_j \frac{c(m_i, \tau_j) - c(m_i, \tau_{j+1})}{\tau_{j+1} - \tau_j} \right)^+, \quad (3)$$

$$p_2(\boldsymbol{\sigma}(\mathbf{m}, \boldsymbol{\tau})) = \sum_{i=1}^{N_m} \sum_{j=1}^{N_\tau} \left(\frac{c(m_{i+1}, \tau_j) - c(m_i, \tau_j)}{m_{i+1} - m_i} \right)^+, \quad (4)$$

$$p_3(\boldsymbol{\sigma}(\mathbf{m}, \boldsymbol{\tau})) = \sum_{i=1}^{N_m} \sum_{j=1}^{N_\tau} \left(\frac{c(m_i, \tau_j) - c(m_{i-1}, \tau_j)}{m_i - m_{i-1}} - \frac{c(m_{i+1}, \tau_j) - c(m_i, \tau_j)}{m_{i+1} - m_i} \right)^+. \quad (5)$$

Static arbitrage constraints [Davis and Hobson, 2007] are then equivalent to

$$\Phi(\boldsymbol{\sigma}(\mathbf{m}, \boldsymbol{\tau})) = 0$$

and the magnitude of $\Phi(\boldsymbol{\sigma}(\mathbf{m}, \boldsymbol{\tau}))$ can be considered as a "distance" from the set of arbitrage-free implied volatility surfaces.

2.2 Dynamics of implied volatility co-movements

Static arbitrage constraints on the shape of the implied volatility surface are a necessary but not sufficient requirement for a good model of implied volatility dynamics: one also needs the model to capture the statistical properties of implied volatility co-movements, a crucial point for any hedging and risk management task. Here we summarise some of the empirically observed statistical properties of implied volatilities on various exchange-traded indices [Cont and da Fonseca, 2002, Avellaneda et al., 2020, Cont and Vuletic, 2023]:

- The implied volatility has a non-flat cross-section, with dependence in strike and maturity.
- Implied volatilities display high positive autocorrelation and mean-reverting behavior.
- Daily variations in the implied volatilities can be satisfactorily explained with a small number of principal components.
- The first principal component corresponds to a *level*, whereas the second principal component corresponds to a *skew* factor.
- The returns of the underlying are negatively correlated with the projections of log-increments of implied volatility on the *level* and *skew* principal components, which is a more precise formulation of the so-called 'leverage effect'.

We now describe a data-driven approach for the simulation of implied volatility dynamics designed to account for the above properties.

3 A generative model for implied volatility surfaces

VOLGAN is a customised conditional generative adversarial network with a smoothness penalty incorporated into the generator's loss function, combined with scenario re-weighting applied to the output scenarios [Cont and Vuletic, 2023].

VOLGAN receives as input

- the implied volatility surface at the previous date,
- the two previous underlying returns,
- the realized volatility from the previous period,

and outputs (joint) scenarios for

- the return of the underlying asset and
- the implied volatility surface

for the next period, along with a set of weights (probabilities) associated with these scenarios. We now discuss the methodology in more detail.

3.1 Architecture

We design a Conditional GAN [Mirza and Osindero, 2014], composed of two neural networks, a *generator* and a *discriminator*. Suppose we have observations at times $t \in \mathbb{T}$, in increments of $\Delta t = 1/252$ (1 day), with S_t the price of the underlying, and $\sigma_t(\mathbf{m}, \boldsymbol{\tau})$ the implied volatility surface on the grid $(\mathbf{m}, \boldsymbol{\tau})$ at time t . Denote by $g_t(\mathbf{m}, \boldsymbol{\tau})$ the log-implied volatility surface at time t :

$$g_t(\mathbf{m}, \boldsymbol{\tau}) = \log \sigma_t(\mathbf{m}, \boldsymbol{\tau}), \quad \Delta g_t(\mathbf{m}, \boldsymbol{\tau}) = g_{t+\Delta t}(\mathbf{m}, \boldsymbol{\tau}) - g_t(\mathbf{m}, \boldsymbol{\tau}). \quad (6)$$

Let R_t be the log-return of the underlying:

$$R_t = \log \left(\frac{S_{t+\Delta t}}{S_t} \right), \quad (7)$$

and denote by γ_t the one-month realized volatility:

$$\gamma_t = \sqrt{\frac{252}{21} \sum_{i=0}^{20} R_{t-i\Delta t}^2}. \quad (8)$$

We aggregate $R_{t-\Delta t}, R_{t-2\Delta t}, \gamma_{t-\Delta t}, g_t(\mathbf{m}, \boldsymbol{\tau})$ into a *condition/input* vector a_t :

$$a_t = (R_{t-\Delta t}, R_{t-2\Delta t}, \gamma_{t-\Delta t}, g_t(\mathbf{m}, \boldsymbol{\tau})). \quad (9)$$

The generator G takes as input this condition a_t and i.i.d. noise $z_t \sim \mathcal{N}(0, I_d)$ and outputs simulated values $\hat{R}_t(z), \Delta \hat{g}_t(\mathbf{m}, \boldsymbol{\tau})$ for the return and implied volatility (log-)increments:

$$G(a_t, z_t) = (\hat{R}_t(z_t), \Delta \hat{g}_t(\mathbf{m}, \boldsymbol{\tau})(z_t)). \quad (10)$$

We denote by $G(a_t, z)|_2 = \Delta \hat{g}_t(\mathbf{m}, \boldsymbol{\tau})(z)$ the second component of the generator's output which corresponds to the simulated log implied volatility increment.

The discriminator is a classifier, taking as input a value $(r, \Delta g)$ representing either the output of the generator or the corresponding data realization, together with a condition vector a_t as in (9). It outputs a value $D(a_t, (R, \Delta g))$ between 0 and 1, interpreted as the probability that the input is drawn from the conditional distribution of $(R_t, \Delta g_t)$ given a_t .

The generator G and the discriminator D are feed-forward neural networks, whose respective parameters (weights) we denote by θ_g and θ_d . The architecture of the generator is displayed in Figure 1, and the architecture of the discriminator is shown in Figure 2.

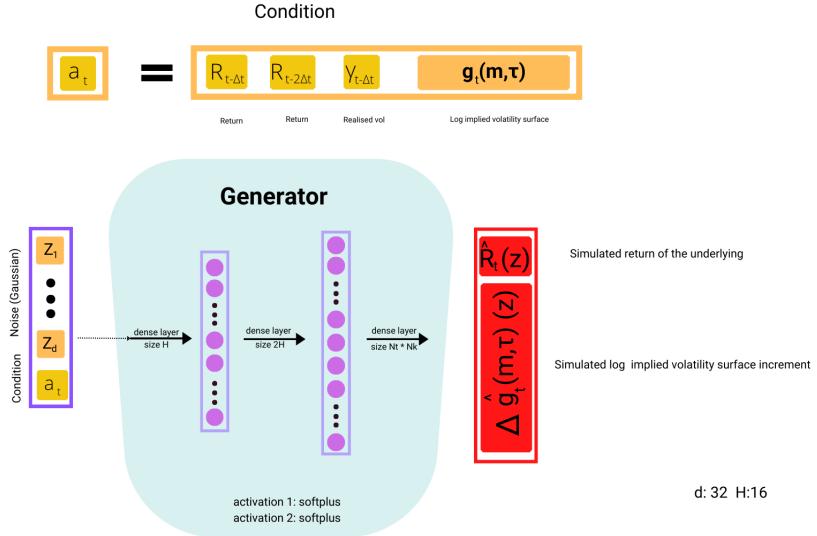


Figure 1: VOLGAN generator architecture.

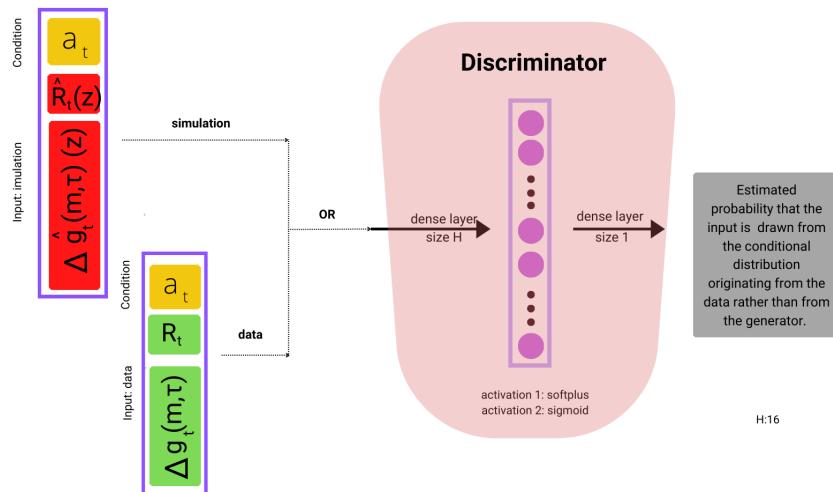


Figure 2: VOLGAN discriminator architecture.

3.2 Training objective

The core component of VOLGAN is a customised loss function catering to the desired properties of the output volatility surface, as advocated in [Cont et al., 2022]. A classical GAN trained using binary cross-entropy (BCE) loss [Goodfellow et al., 2014] would result in irregular surfaces. In order to generate smooth surfaces, we use a smoothness penalty [Sana and Cont, 2005, Jackson et al., 1999] defined as a discrete Sobolev semi-norm in m and τ on the grid $(\mathbf{m}, \boldsymbol{\tau})$:

$$L_m(\mathbf{g}) = \sum_{i,j} \frac{(\mathbf{g}(m_{i+1}, \tau_j) - \mathbf{g}(m_i, \tau_j))^2}{|m_{i+1} - m_i|^2} \simeq \|\partial_m g\|_{L^2}^2, \quad (11)$$

$$L_\tau(\mathbf{g}) = \sum_{i,j} \frac{(\mathbf{g}(m_i, \tau_{j+1}) - \mathbf{g}(m_i, \tau_j))^2}{|\tau_{j+1} - \tau_j|^2} \simeq \|\partial_\tau g\|_{L^2}^2. \quad (12)$$

These terms are included in the training objective $J^{(G)}(\theta_d, \theta_g)$ for the generator:

$$\begin{aligned} J^{(G)}(\theta_d, \theta_g) = & -\frac{1}{2} \mathbb{E} [\log (D(a_t, G(a_t, z_t; \theta_g); \theta_d))] \\ & + \alpha_m \mathbb{E} [L_m(g_t(\mathbf{m}, \boldsymbol{\tau}) + G(a_t, z_t; \theta_g)|_{2:})] \\ & + \alpha_\tau \mathbb{E} [L_\tau(g_t(\mathbf{m}, \boldsymbol{\tau}) + G(a_t, z_t; \theta_g)|_{2:})], \end{aligned} \quad (13)$$

where $a_t = (R_{t-\Delta t}, R_{t-2\Delta t}, \gamma_{t-\Delta t}, g_t(\mathbf{m}, \boldsymbol{\tau}))$, as defined in (9). The first term is a binary cross-entropy for the output of the discriminator. $\alpha_m > 0$ and $\alpha_\tau > 0$ are regularisation parameters, a_t is the input "condition" (Eq. (9)); θ_g and θ_d are respectively the parameters (weights) of the generator and the discriminator networks. The expectation is computed over the law of the IID (Gaussian) input $\mathbf{z}_t \sim N(0, I_d)$. The smoothness penalties L_m and L_τ are applied to the simulated log-implied volatility surfaces:

$$g_t(\mathbf{m}, \boldsymbol{\tau}) + G(a_t, z_t; \theta_g)|_{2:} = g_t(\mathbf{m}, \boldsymbol{\tau}) + \Delta \hat{g}_t(\mathbf{m}, \boldsymbol{\tau})(z_t) = \hat{g}_t(\mathbf{m}, \boldsymbol{\tau})(z_t).$$

The discriminator is trained to minimise the binary cross-entropy loss:

$$\begin{aligned} J^{(D)}(\theta_d, \theta_g) = & -\frac{1}{2} \mathbb{E} [\log (D(a_t, (R_t, \Delta g_t(\mathbf{m}, \boldsymbol{\tau})); \theta_d))] \\ & - \frac{1}{2} \mathbb{E} [\log (1 - D(a_t, G(a_t, z_t; \theta_g); \theta_d))], \end{aligned} \quad (14)$$

where a_t is the input condition (Eq. (9)), R_t and $\Delta g_t(\mathbf{m}, \boldsymbol{\tau})$ are the corresponding data.

We assume the process $(R_t, g_t)_{t \geq 0}$ to be ergodic, so given a long enough sample $t \in \mathbb{T}$ we can approximate the expected values above by sample averages:

$$\mathbb{E}[f(R_t, g_t)] \simeq \frac{1}{|\mathbb{T}|} \sum_{t \in \mathbb{T}} f(R_t, g_t).$$

It is possible to incorporate the arbitrage penalty (2) into the loss function of the generator (13). However, we have not done so, and our numerical experiments indicate no notable difference when including it, suggesting that the smoothness penalty is enforcing shape constraints indirectly.

3.3 Scenario re-weighting

The outputs of the generator described above are not guaranteed to satisfy the static arbitrage constraints described in Section 2.1. To correct for this, we apply the methodology described in [Cont and Vuletic, 2023] to re-weight the one-day-ahead scenarios generated by the GAN.

Let \mathbb{P}_0 be the law of the generator's output i.e. the joint dynamics of the underlying return and the implied volatility surface $(R_t, \sigma_t(\mathbf{m}, \boldsymbol{\tau}); t \in \mathbb{T})$. To adjust for static arbitrage, [Cont and Vuletic, 2023] apply the change of measure:

$$\frac{d\mathbb{P}_\beta}{d\mathbb{P}_0}(\omega) = \frac{\exp(-\beta\Phi(\sigma(\mathbf{m}, \boldsymbol{\tau}; \omega)))}{Z(\beta)} \quad (15)$$

where $Z(\beta)$ is a normalization factor:

$$Z(\beta) = \mathbb{E}^{\mathbb{P}_0} [\exp(-\beta\Phi(\sigma(\mathbf{m}, \boldsymbol{\tau}; \omega)))] \quad (16)$$

VOLGAN samples from this target distribution (15) using a Weighted Monte Carlo approach. Given N samples from the generator $(\hat{R}^i, \hat{\boldsymbol{\sigma}}^i)$, $i = 1, \dots, N$, we compute the arbitrage penalty $\Phi(\hat{\boldsymbol{\sigma}}^i)$ corresponding to each output scenario $(\hat{R}^i, \hat{\boldsymbol{\sigma}}^i)$ using (2) and sample the scenario $(\hat{R}^i, \hat{\boldsymbol{\sigma}}^i)$ with probability

$$w^i = \frac{\exp(-\beta\Phi(\hat{\boldsymbol{\sigma}}^i))}{\sum_{j=1}^N \exp(-\beta\Phi(\hat{\boldsymbol{\sigma}}^j))}. \quad (17)$$

These weighted scenarios may then be used to compute expectations and quantiles of various quantities of interest under \mathbb{P}_β . Let X be a function of the state variables, and let x_i be its value in scenario i . Denote by $F_{X,\beta}$ the law of X under \mathbb{P}_β and by $\mathbb{E}_\beta[X]$ its expectation. We can estimate $\mathbb{E}_\beta[X]$ by

$$\widehat{\mathbb{E}_\beta[X]} = \sum_{i=1}^N w_i x_i, \quad (18)$$

while the quantiles of X are estimated as

$$\widehat{F_{X,\beta}^{-1}(q)} = x_{(k)}, \quad \text{where } k = \min\{j \in \{1, \dots, N\} : \sum_{i=1}^j w_{(i)} \geq q\}, \quad (19)$$

where $x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(N)}$ are the order statistics of x_1, \dots, x_N .

3.4 Numerical implementation

The generator G is a three-layer feedforward dense neural network, with the first two activations softplus, and the final layer an affine layer. The random input is (standard) i.i.d Gaussian noise with dimension $d = 32$. The first layer consists of $H = 16$ neurons, whereas the second layer contains $2H = 32$ neurons. Similarly, the discriminator D is a two-layer feedforward neural network, with softplus and sigmoidal activation functions and layer sizes of $H = 16$ and 1, respectively. The discriminator has a simpler architecture than the generator, as it is of the utmost importance to keep the two neural networks in balance. The architecture of the discriminator is shown in Figure 2, and the architecture of the generator is displayed in Figure 1.

The hyperparameters $\alpha_m, \alpha_\tau > 0$ are chosen by *gradient norm matching*. We first train VOLGAN for $n_{grad} = 25$ epochs by performing optimisation via the binary cross-entropy loss only (classical GAN setting). At each update, we calculate the gradient norms of each of the three loss function terms in (13): BCE, L_m , L_τ with respect to θ_g . We then set α_m and α_τ , to be the means of observed ratios of the gradient norms of the BCE term to the gradient norms of the L_m and L_τ , respectively. The gradient norms of the BCE, L_m , L_τ terms with respect to θ_g during this stage are shown in Figure 3. We note that all three gradients behave similarly, that they stabilise over time, and that there is no gradient explosion or vanishing gradient phenomena.

We then restart training VOLGAN (from the same initialisation used for the start of the gradient norm matching procedure) with the loss function defined by Equation (13) for $n_{epochs} = 10000$ epochs, using an alternating direction method i.e. one discriminator update for each generator update. The optimiser used is RMSProp [Hinton et al., 2012], and the learning rates of both networks are set to 0.0001. We take $N = 10000$ raw samples from the generator. The mini-batch size is $n_{batch} = 100$.

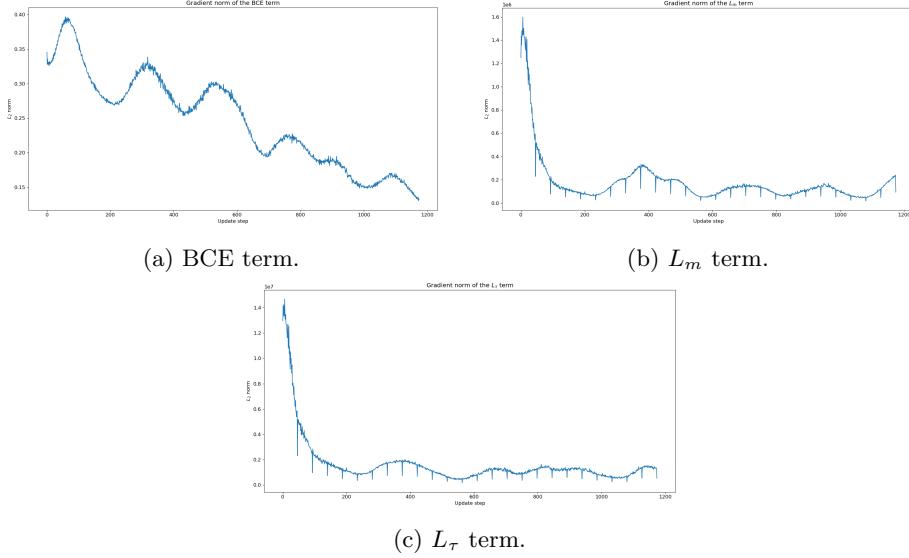


Figure 3: Norm of gradient of the BCE term, L_m term, and L_τ term with respect to θ_g during the first stage of VolGAN training. n

Calibration of β The hyperparameter β might be chosen by considering the Kullblack-Leibler divergence between the distribution of the weights and the uniform distribution on the scenarios [Cont and Vuletic, 2023]. Based on the results in [Cont and Vuletic, 2023], we set

$$\beta(t) = \frac{500}{\max\{w_i(t)\}}, \quad (20)$$

where $w_i(t)$ are the weights associated with the generator outputs on day t .

4 Learning to simulate SPX implied volatility surfaces

To demonstrate VOLGAN’s ability to generate realistic scenarios for SPX implied volatility dynamics, we train VOLGAN on the daily time series of market data and examine the properties of the generator thus trained. The same approach might be applied to other equity options.

4.1 Data

We use the Option Prices file from OptionMetrics. The time period in question is from the 3rd January 2000 to the 28th February 2023, with 3rd Jan 2000-16th Jun 2018 corresponding to the training, and 17th Jun 2019-28th Feb 2023 to the test set. The historical VIX closing prices are available on the CBOE website. The implied risk-free interest rate for each day is calculated as the median rate implied by the put-call parity from the option mid-prices. We construct smooth implied volatility surfaces using the kernel smoothing methodology of [Cont and da Fonseca, 2002, OptionMetrics, 2021]. Our grid (\mathbf{m}, τ) consists of $m \in \{0.6, 0.7, 0.8, 0.9, 0.95, 1, 1.05, 1.1, 1.2, 1.3, 1.4\}$ and of times to maturity $\tau \in \{\frac{1}{252}, \frac{1}{52}, \frac{2}{52}, \frac{1}{12}, \frac{1}{6}, \frac{1}{4}, \frac{1}{2}, \frac{1}{4}, 1\}$, one day to one year. Suppose that on a fixed day we have available implied volatility data $\sigma(m, \tau)$ for $m \in \mathcal{M}$ and $\tau \in \mathcal{T}$, with corresponding values of Vega $\kappa(m, \tau)$. We consider a Vega-weighted Nadaraya-Watson kernel smoothing estimator with a 2D Gaussian kernel:

$$\hat{\sigma}(m', \tau') = \frac{\sum_{m \in \mathcal{M}, \tau \in \mathcal{T}} \kappa(m, \tau) k(m - m', \tau - \tau') \sigma(m, \tau)}{\sum_{m \in \mathcal{M}, \tau \in \mathcal{T}} \kappa(m, \tau) k(m - m', \tau - \tau')}, \quad (21)$$

where:

$$k(x, y) = \frac{1}{2\pi} \exp \left[-\frac{x^2}{2h_1} - \frac{y^2}{2h_2} \right].$$

In order to determine the values of the bandwidth hyperparameters h_1 and h_2 , we sample a day uniformly at random from the first 100 days available (which was 31st Jan 2000) and find the pair of hyperparameters (h_1, h_2) minimizing the arbitrage penalty. We conduct the search over values between 0.002 and 0.1 (inclusive) in 0.002 increments, for both h_1 and h_2 . The minimizer of the arbitrage penalty was the pair $(h_1, h_2) = (0.002, 0.046)$. The resulting arbitrage penalty over the entire data set after smoothing is shown in Figure 4. Note that compared to [Cont and Vuletic, 2023] we include shorter times to maturity and use a different dataset.

To simplify the notation, we will use $\sigma_t(\mathbf{m}, \tau)$ for the implied volatility surface obtained after smoothing, on the (\mathbf{m}, τ) grid. For general $\sigma_t(m, \tau)$ we interpolate $\sigma_t(\mathbf{m}, \tau)$ linearly first in moneyness, and then in time to maturity. When extrapolation is necessary, it is linear.

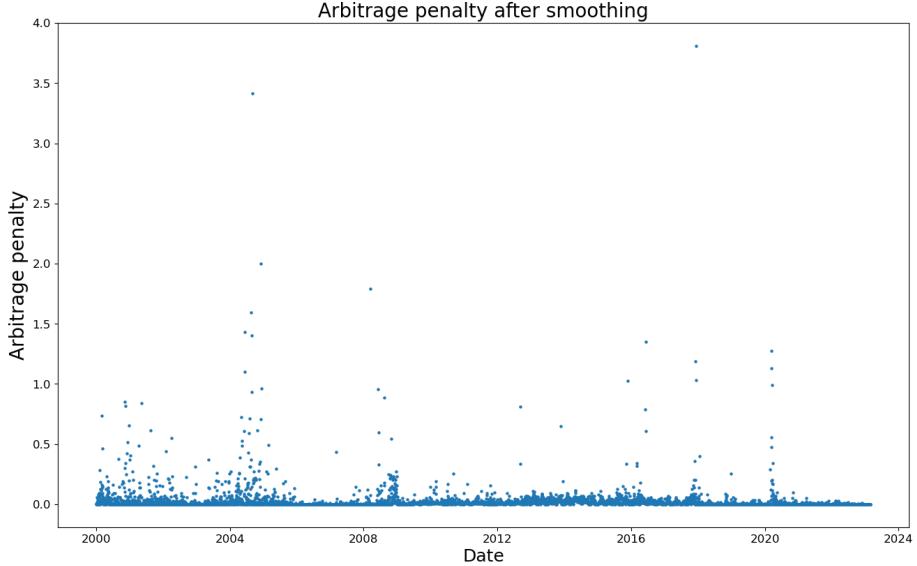


Figure 4: Arbitrage penalty for SPX implied volatility surface after smoothing.

4.2 Out-of-sample performance

As discussed in Section 2, the main goal of an implied volatility model is to correctly capture the co-movements of implied volatilities, while satisfying static arbitrage constraints. We can measure the latter by considering the 'distance to arbitrage' using the arbitrage penalty (2). In order to measure how well VOLGAN learns the dynamics and captures the co-movements of implied volatilities, we perform PCA on the generated increments, and compare them with the principal components of the data increments. Furthermore, we simulate the CBOE volatility index VIX [CBOE, 2022], which is a non-linear combination of tradable calls and puts. We compare the dynamics of the simulated and market data.

4.2.1 Detecting extreme market events

Firstly, we note that the trained discriminator might be used for detecting extreme market events. Figure 5 contains discriminator scores on the training and testing data. Since the discriminator has already been trained, it is of no surprise that the outputs cluster around 0.5. There are two clusters of points with scores lower than others: those corresponding to the 2008 financial crisis (in-sample) and to the start of the Covid-19 pandemic (out-of-sample). In particular, the discriminator assigns a score below 0.2 to the data from the start of the Covid-19 pandemic, highlighting the difference in this data compared to the rest of the training and test set.

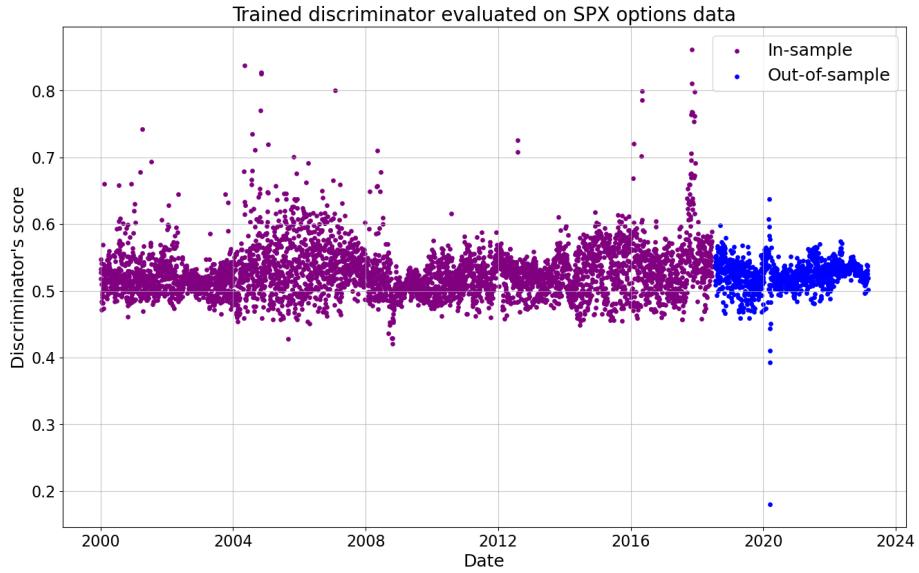


Figure 5: Discriminator output score on in-sample and out-of-sample SPX options data.

4.2.2 Smoothness and arbitrage constraints

Incorporating the smoothness penalty (11)-(12) into the loss function (13) is crucial for generating smooth surfaces. As shown in Figure 6, training via the classical Binary Cross-Entropy (BCE) loss [Goodfellow et al., 2014], using the same architecture, hyperparameters, and the same number of training epochs, results in irregular surfaces.

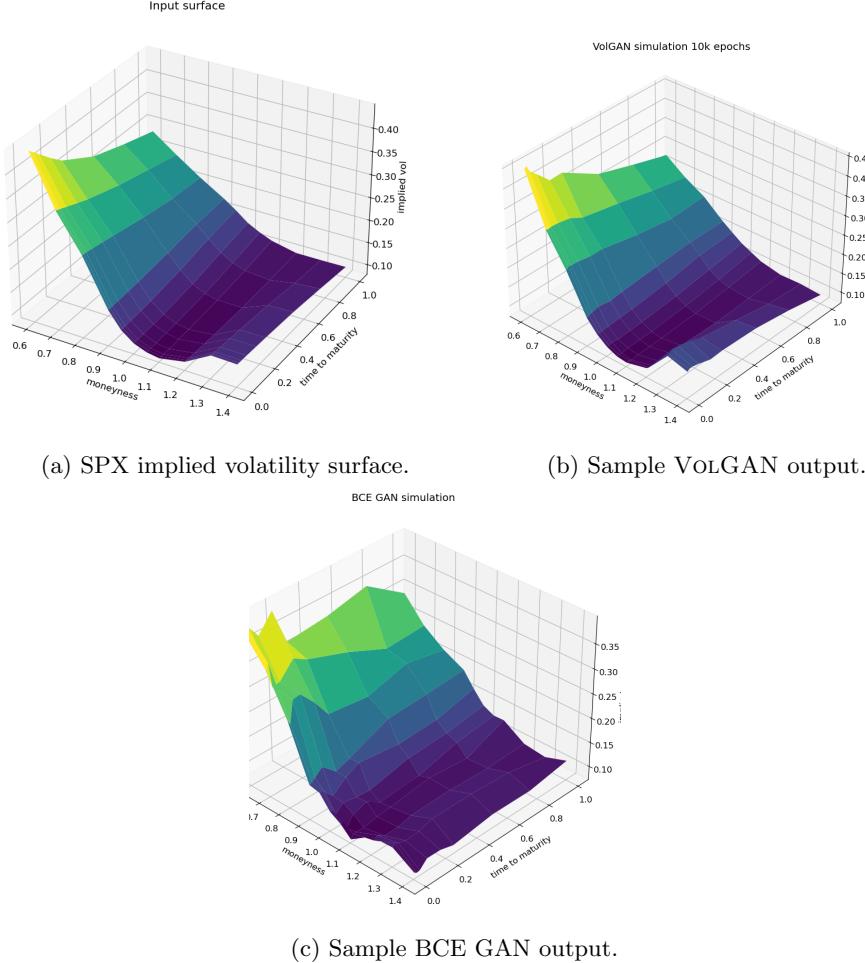


Figure 6: Implied volatility surfaces generated using (b) VOLGAN (c) classical GAN, compared with (a) SPX implied volatility surface.

As the input surfaces might admit static arbitrage, it is not realistic to expect outputs to be completely arbitrage-free. What is plausible, however, is for the outputs to have arbitrage penalties of the same order (or lower) than the inputs. Table 1 compares out-of-sample arbitrage penalties for SPX implied volatilities and the outputs of the BCE GAN and VOLGAN with/without scenario re-weighting. Arbitrage penalties in the BCE GAN samples are observed to be high: this is linked to the previous observation that BCE GAN fails to generate smooth surfaces, resulting in failure of static arbitrage conditions which are linked to derivatives of the surfaces. In contrast, VOLGAN outputs have arbitrage penalty levels similar to the input data. Scenario re-weighting leads to a low probability of selecting scenarios with static arbitrage, as shown in Figure 7, where the

reduction in arbitrage is visualized. The mean, standard deviation, and median values from Table 1 correspond to the statistics of the time series displayed in Figure 7. We note that during 2022 there is more volatility in arbitrage penalty in VOLGAN compared to the remainder of the test period

	Mean	Std	Median
Market data	0.0096	0.0628	0.0005
BCE GAN	2.4635	0.9086	2.3164
Raw VolGAN (before weighting)	0.0199	0.088	0.003
VolGAN (after re-weighting)	0.0127	0.0620	0.0014

Table 1: Arbitrage penalties in SPX implied volatility market data (test set) vs generated data via GANs trained using (i) BCE loss only (ii) VOLGAN loss (iii) VOLGAN re-weighted scenarios (adaptive β). Standard deviation and median for GAN outputs correspond to the standard deviation and the median of (re-weighted) average outputs given 10000 samples.

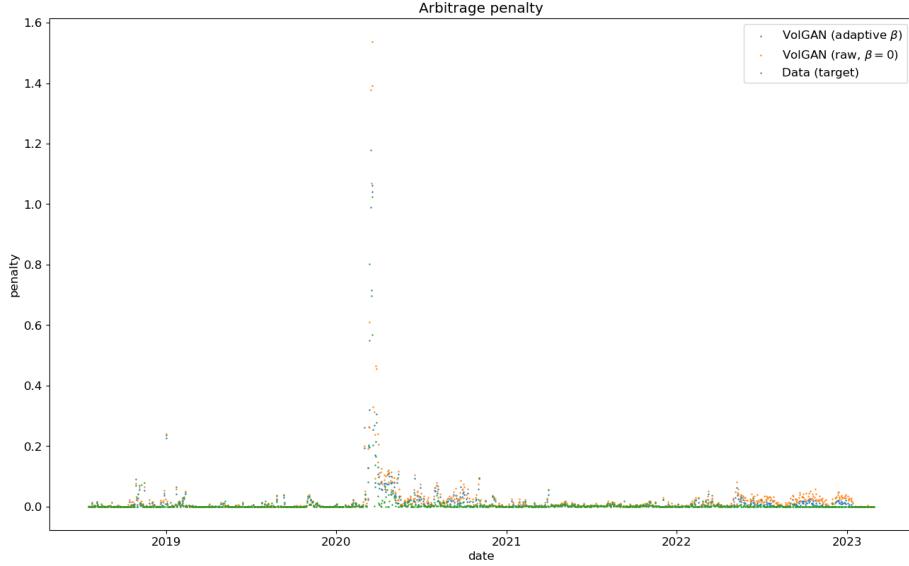


Figure 7: Distance to arbitrage as measured by the arbitrage penalty (2) in SPX implied volatility data (red) vs. mean arbitrage penalty of surfaces generated via VOLGAN, before (blue) and after (green) scenario re-weighting.

4.2.3 Next-day forecasting

We use VOLGAN to generate next-day forecasts using the conditional expectation of the variable given the history, together with a 95% confidence interval obtained by considering the 2.5% and 97.5% quantiles for the following quantities of interest:

- index level S_t ;
- VIX level σ_t^{VIX} ;
- a range of implied volatilities $\sigma_t(m, \tau)$ with

$$\tau \in \left\{ \frac{1}{252}, \frac{1}{52}, 0.25, 0.125 \right\}, \quad m \in \{0.75, 1, 1.25\}$$

Figures 8, 9, 10, 11 compare respectively the 3-month, 1-month, 1-week, and 1-day ATM implied volatility with the VOLGAN one-day ahead 95% confidence interval forecast, displaying good agreement with observations. VOLGAN appears to slightly overestimate implied volatility levels for $m > 1$ but not for $m < 1$, as shown in Figures 12 and 13.

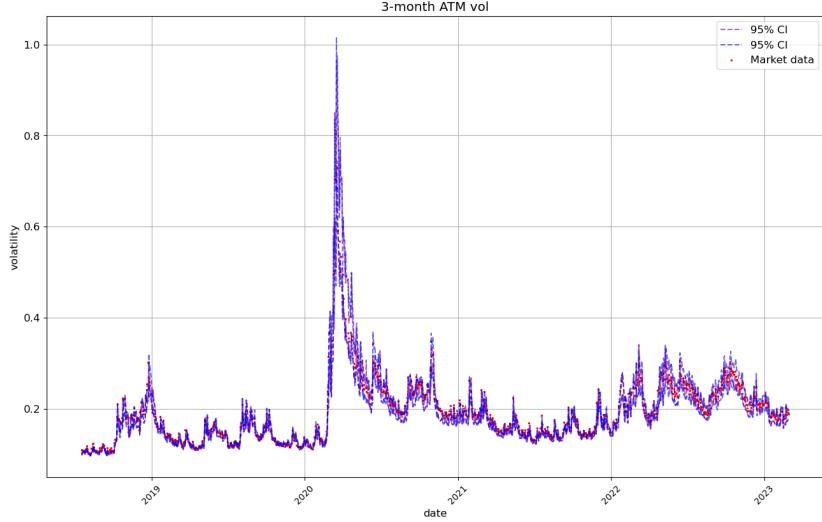


Figure 8: 3-month ATM implied volatility ($m = 1, \tau = 0.25$): market data (red), next-day forecast ($\mathbb{E}_\beta[\sigma_t(1, 0.25) | a_{t-\Delta t}]$) and 95% confidence interval (blue) based on the 2.5% and 97.5% VOLGAN quantiles.

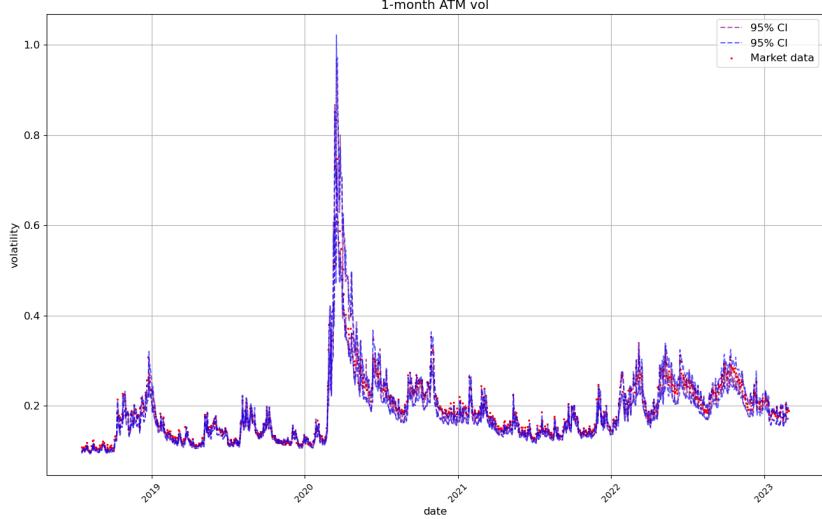


Figure 9: 1-month ATM implied volatility ($m = 1, \tau = 1/12$): market data (red), next-day forecast ($\mathbb{E}_\beta[\sigma_t(1, 1/12) | a_{t-\Delta t}]$) and 95% confidence interval (blue: without re-weighting, purple: with re-weighting) based on the 2.5% and 97.5% VOLGAN quantiles.

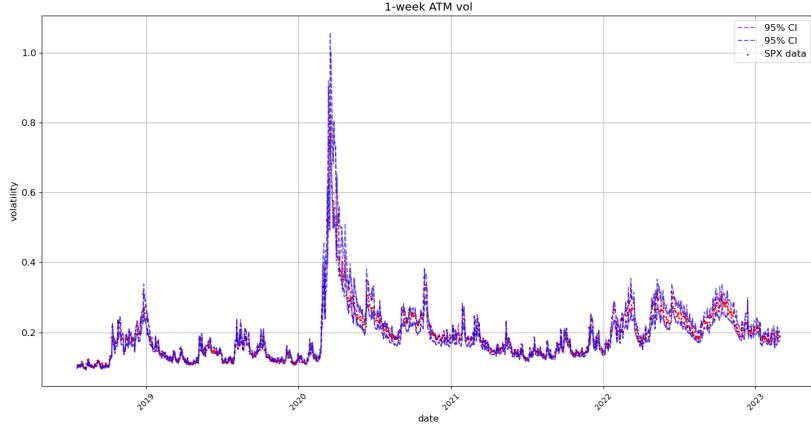


Figure 10: 1-week ATM implied volatility ($m = 1, \tau = 1/52$): market data (red), next-day forecast ($\mathbb{E}_\beta[\sigma_t(1, 1/52) | a_{t-\Delta t}]$) and 95% confidence interval (blue: without re-weighting, purple: with re-weighting) based on the 2.5% and 97.5% VOLGAN quantiles.

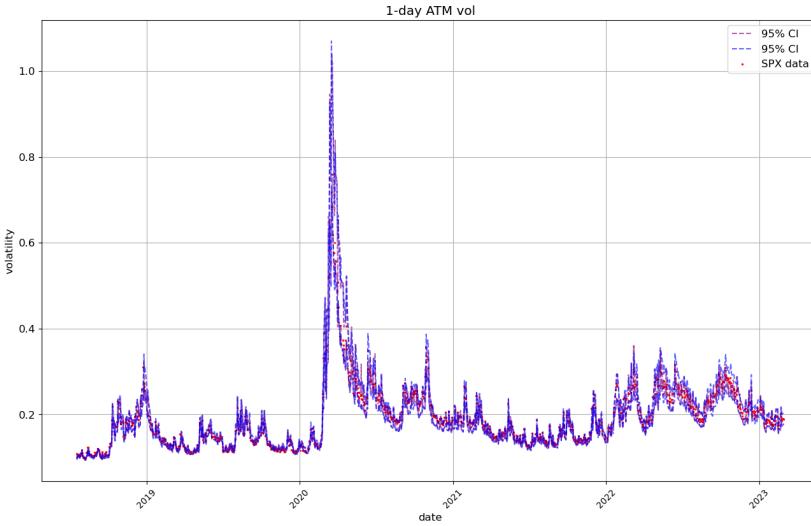


Figure 11: 1-day ATM implied volatility ($m = 1, \tau = 1/252$): market data (red), next-day forecast ($\mathbb{E}_\beta[\sigma_t(1, 1/252) | a_{t-\Delta t}]$) and 95% confidence interval (blue: without re-weighting, purple: with re-weighting) based on the 2.5% and 97.5% VOLGAN quantiles.

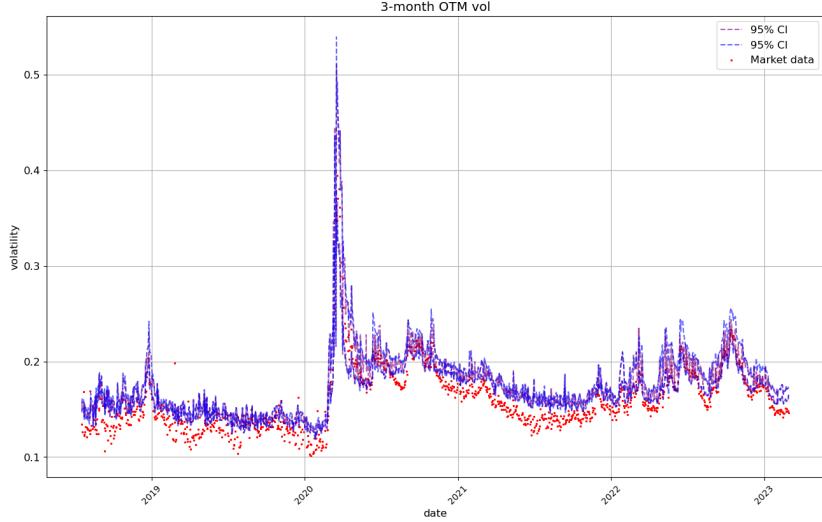


Figure 12: 3-month OTM call implied volatility ($m = 1.25, \tau = 0.25$): market data (red), next-day forecast ($\mathbb{E}_\beta[\sigma_t(1.25, 0.25)|a_{t-\Delta t}]$) and the 95% confidence interval (blue: without re-weighting, purple: with re-weighting). The confidence interval is calculated based on the 2.5% and 97.5% VOLGAN quantiles.

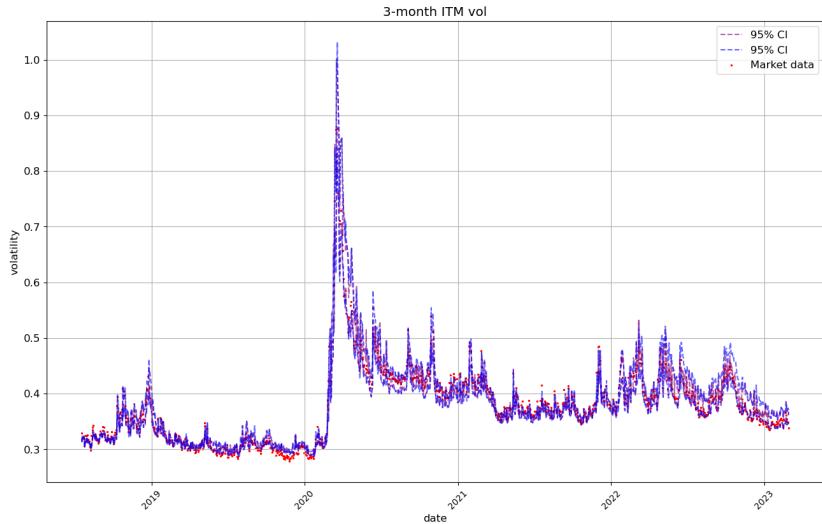


Figure 13: 3-month ITM call implied volatility ($m = 0.75, \tau = 0.25$): market data (red), next-day forecast ($\mathbb{E}_\beta[\sigma_t(0.75, 0.25)|a_{t-\Delta t}]$) and 95% confidence interval (blue: without re-weighting, purple: with re-weighting). The confidence interval is calculated based on the 2.5% and 97.5% VOLGAN quantiles.

Figure 14 displays the simulated and real SPX returns, showing that VOLGAN confidence intervals appropriately capture the underlying. We visualize the impact of scenario re-weighting on the confidence intervals in Figure 15. During periods of high arbitrage penalty, a small number of simulations hold most of the weight, therefore inducing very narrow confidence intervals. This behaviour is visible not just in the simulations for the underlying, but for the ATM ($m = 1$), OTM ($m = 0.75$), and ITM ($m = 1.25$) implied volatilities (Figures 8, 12, 13 respectively). From Figure 15, we note that if arbitrage is not penalized ($\beta = 0$), the forecasts are more accurate, including for March and April 2020. However, choosing to use the raw generator might result in static arbitrage of the mid-prices. As before, we note that the width of the confidence intervals varies with time, with the confidence intervals appearing more consistent in 2022. The raw generator ($\beta = 0$) produces stable confidence intervals for all state variables, highlighting VOLGAN’s stability and not requiring frequent re-calibration.

Figure 16 compares one-day ahead simulated values of VIX, computed from its definition in terms of simulated call/put prices, with the VIX closing prices on target days in the test set. VOLGAN simulations are on the same scale as VIX. Some of the differences might be coming from the discrete approximation of the log-contract used for computation of simulated VIX values [CBOE, 2022].

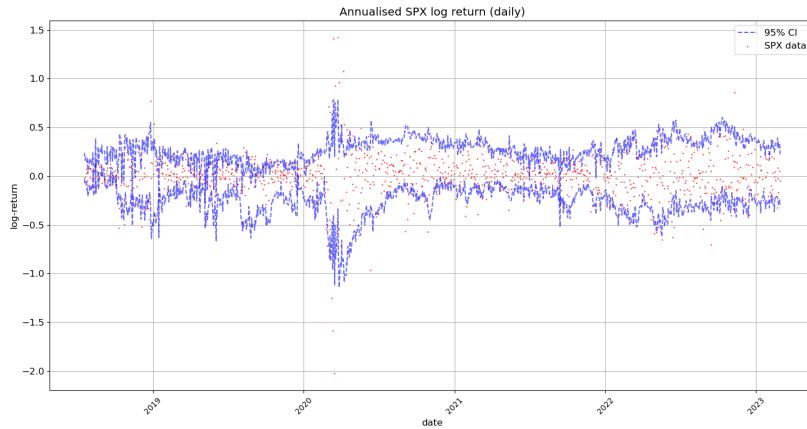


Figure 14: Realized and simulated SPX log-return on the test set. Market data (red), next-day forecast ($\mathbb{E}_\beta[S_t|a_{t-\Delta t}]$) and the 95% confidence interval (blue: without re-weighting). The confidence interval is calculated based on the 2.5% and 97.5% VOLGAN quantiles.

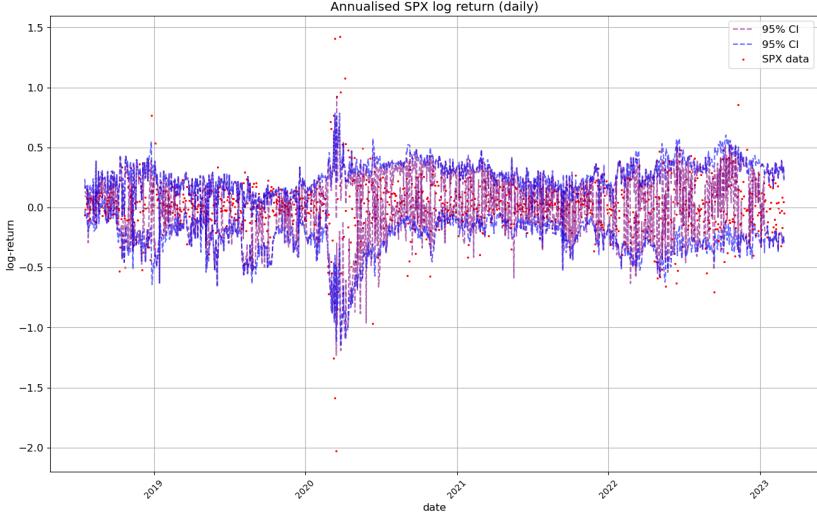


Figure 15: Realized and simulated SPX log-return on the test set. Market data (red), next-day forecast ($\mathbb{E}_\beta[S_t|a_{t-\Delta t}]$) and the 95% confidence interval (blue: without re-weighting, purple: with re-weighting). The confidence interval is calculated based on the 2.5% and 97.5% VOLGAN quantiles before and after re-weighting.

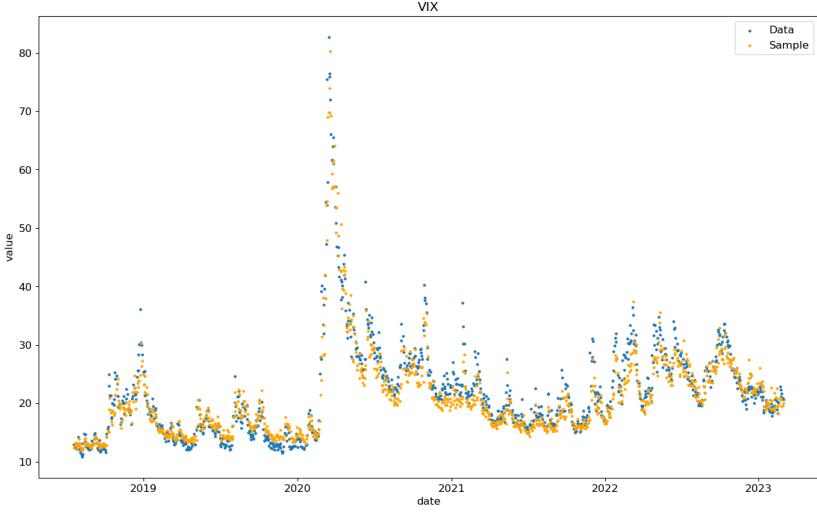


Figure 16: Historical vs one-day ahead simulation of VIX, on test data set.

We further investigate the prediction score in Table 2 by considering the percentage of data realizations falling below the simulated 1%, 2.5%, 97.5%, and

99% quantiles. We note that the best overall forecasts are for the underlying. VOLGAN underestimates extremely high values of the implied volatility returns and VIX. Given that the volatility index is a non-linear transformation of the state variables, it is not surprising that VOLGAN does not produce as stable confidence intervals as it does for the state variables. The findings from Table 2 are in line with the previous observations: VOLGAN captures the state variables for which more data is available better. It is important to note that the observed behaviour is out-of-sample, four and a half years after training, including the 2020 data.

Variable/Quantile	0.01	0.025	0.975	0.99
SPX return	25.32%	29.19%	82.00%	83.55%
3-month ATM vol	13.95%	15.16%	49.61%	54.61%
3-month OTM vol	76.978%	78.81%	92.85%	93.80%
3-month ITM vol	29.46%	30.32%	65.46%	69.34%
1-month ATM vol	9.82%	11.28%	42.89%	48.41%
1-week ATM vol	20.41%	22.05%	59.17%	63.22%
1-day ATM vol	19.90%	21.79%	60.12%	64.34%
VIX	34.37%	35.23%	52.67%	55.04%

Table 2: Exceedance ratio for VOLGAN quantiles on the test set.

As already observed in Figure 15, there are instances (of market turbulence) where *not* correcting for the presence of static arbitrage (i.e. setting $\beta = 0$) actually *improves* forecasting performance. We note that when the arbitrage penalty is very low or zero, the penalization has negligible impact on the simulated confidence intervals. Table 2 shows that choosing $\beta = 0$ can in fact improves forecasts, especially for SPX returns, 1-week ATM volatility, and VIX.

Variable/Quantile	0.01	0.025	0.975	0.99
SPX return	4.48%	9.39%	92.33%	93.37%
3-month ATM vol	8.52%	9.56%	64.51%	71.67%
3-month OTM vol	72.18%	73.64%	97.59%	98.02%
3-month ITM vol	20.33%	22.14%	75.62%	81.83%
1-month ATM vol	5.25%	6.55%	57.88%	66.58%
1-week ATM vol	11.80%	13.78%	72.95%	80.10%
1-day ATM vol	11.71%	13.52%	74.68%	81.65%
VIX	25.24%	25.84%	71.23%	71.18%

Table 3: Exceedance ratio for VOLGAN quantiles on test set with $\beta = 0$.

4.2.4 Distributions and correlations learned by the generator

Denote by ρ_t the instantaneous correlation between the 1-month ATM volatility returns and the returns of the underlying at time t . We would like to explore whether or not VOLGAN learns constant correlations. Therefore, we perform the following hypothesis test:

$$H_0: \rho_t = \rho \text{ is constant,} \quad H_1: \rho_t \neq \rho \text{ is time-varying.}$$

Under H_0 , the 95% confidence interval for ρ_t is given by $[\rho^L, \rho^U]$, where [Bonett and Wright, 2000]

$$\rho^U = \frac{\exp(2z_U) - 1}{\exp(2z_U) + 1}, \quad \rho^L = \frac{\exp(2z_L) - 1}{\exp(2z_L) + 1};$$

$$z_U = \frac{1}{2} \log \left[\frac{1 + \rho}{1 - \rho} \right] + \sqrt{\frac{1}{n - 3}} z_{0.975}, \quad z_L = \frac{1}{2} \log \left[\frac{1 + \rho}{1 - \rho} \right] - \sqrt{\frac{1}{n - 3}} z_{0.975},$$

where n is sample size. Estimating ρ by the sample mean of ρ_t on the test set, in Figure 17 we plot ρ_t and the 95% confidence interval $[\rho^L, \rho^U]$. We note that ρ_t is away from the confidence interval of H_0 , indicating strong evidence against H_0 . VOLGAN learns time-varying instantaneous correlations which would be difficult to capture with a parametric model.

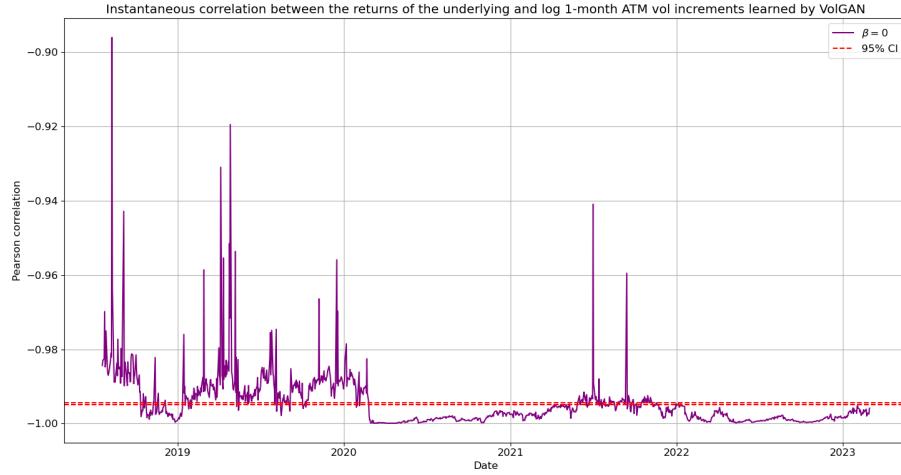


Figure 17: Pearson correlation between simulated index returns and 1-month ATM volatility increments (blue), with symmetric 95% confidence interval of constant correlation (red). VOLGAN with $\beta = 0$.

We compare the (simulated) distributions of the daily returns for the underlying and 1-month ATM volatility with the corresponding empirical distributions and with Gaussian distributions with the same mean and variance. Figures 18 and 19 shows that simulated index returns and ATM volatility increments have asymmetric, non-Gaussian and exponentially decaying tails. Such non-Gaussian, asymmetric distributions are difficult to capture in a model with Brownian increments.

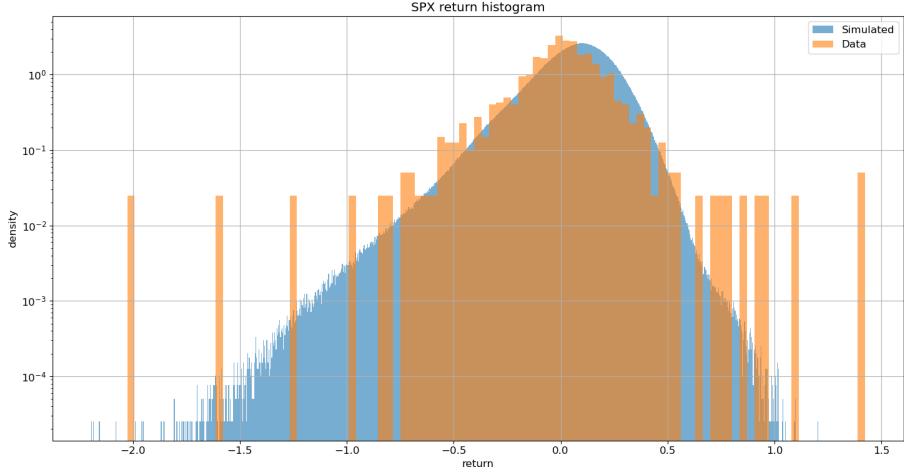


Figure 18: Simulated index returns (blue) exhibit asymmetric, exponentially decaying tails. VOLGAN with $\beta = 0$.

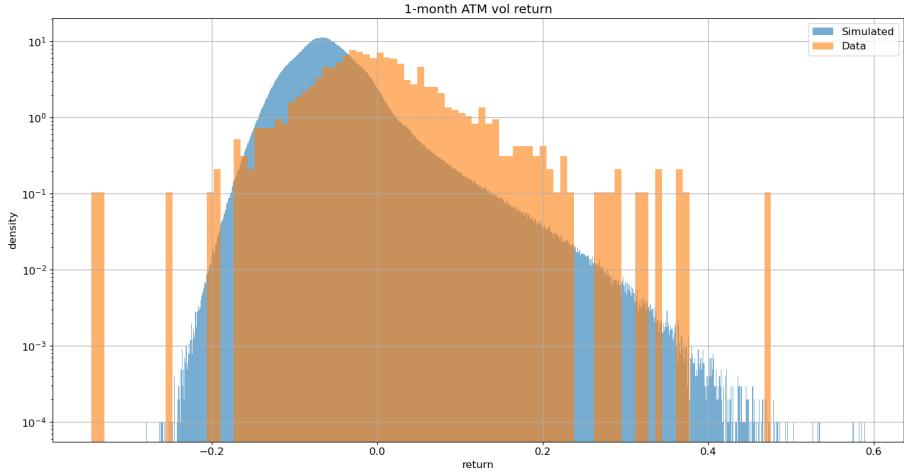


Figure 19: Simulated 1-month ATM volatility increments (blue) exhibit asymmetric, exponentially decaying tails. VOLGAN with $\beta = 0$.

4.2.5 Principal component analysis

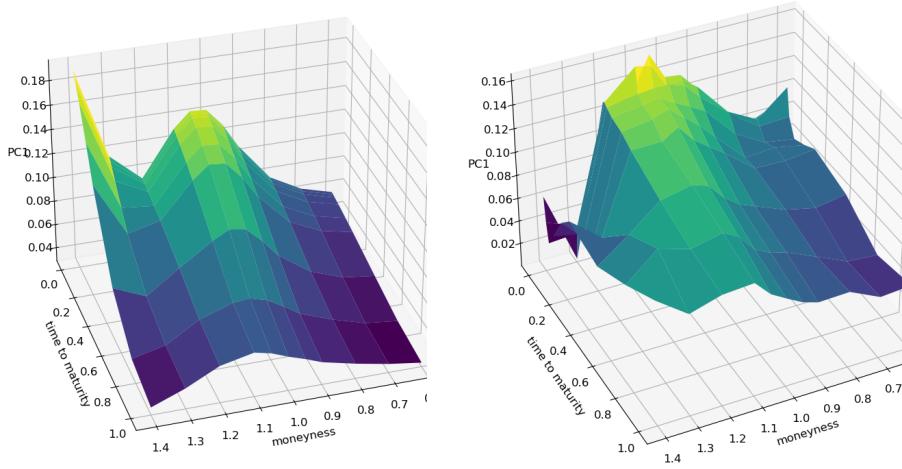
In order to investigate VOLGAN’s ability to appropriately capture the implied volatility co-movements, we perform out-of-sample principal component analysis on the simulated log increments of implied volatility. We compare the first three simulated principal components with the corresponding PCs of the data

realizations. When performing PCA on four and a half years of SPX implied volatility data, the eigenvectors change depending on the period of observation, but nonetheless correspond to *level*, *skew* and *curvature*. In Table 4 we show variance explained by the first three eigenvectors in the testing data and in the VOLGAN simulations. The significance of the first two principal components is very similar in the test data and in VOLGAN. The third principal component is more significant in the simulated data compared to the market data.

Rank	Data	VOLGAN
First	51.25%	$45.31 \pm 1.84\%$
Second	34.00%	$25.69 \pm 0.88\%$
Third	5.01%	$12.76 \pm 0.55\%$

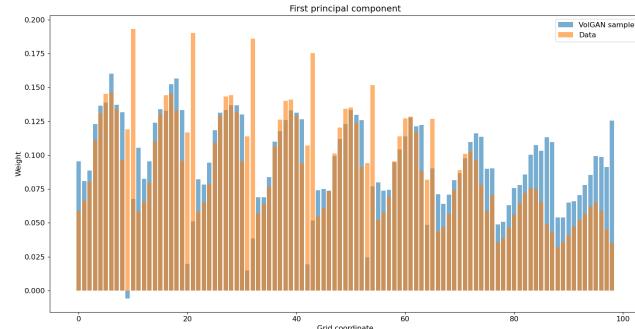
Table 4: Out-of-sample (two years after training) percentage of variance explained by the top three principal components of the simulated and the data log implied volatility increments. The VOLGAN column contains the average $\pm 1.96 \times$ standard deviation of the observed values, across 1000 VoLGAN samples.

The first principal components of the sample VOLGAN implied volatility log-returns and of the corresponding SPX implied volatility market data are displayed in Figure 20. Both surfaces are consistently positive, indicating that they might have a *level* interpretation. The second eigenvectors of both SPX implied volatility market data and of the simulated scenarios (Figure 21) can be interpreted as *skew*, while the third eigenvectors (Figure 22) can be interpreted as *curvature*. Figures 20, 21, 22 reflect on the clear resemblance between the principal components of the SPX implied volatility market data and of the VOLGAN simulations, showing that VOLGAN is able to dynamically learn the covariance structure of implied volatility co-movements.



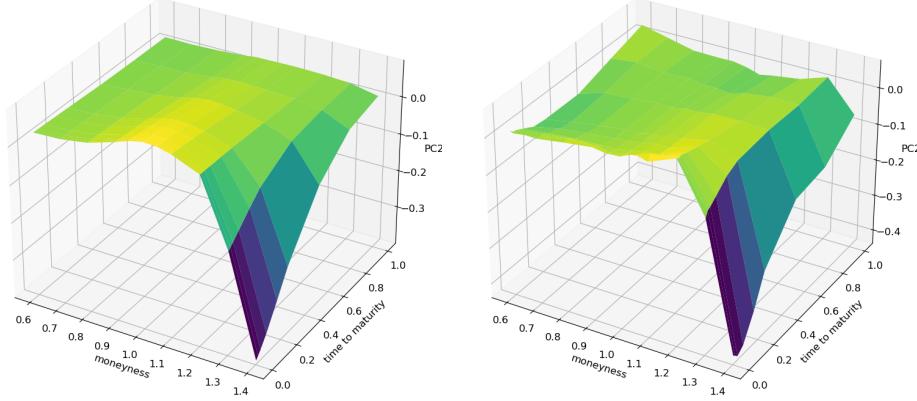
(a) Computed using SPX implied volatility data.

(b) Computed using a sample VolGAN output.



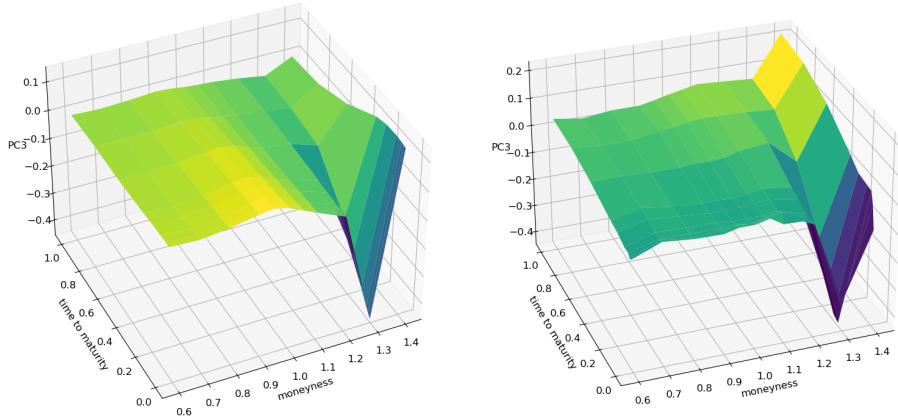
(c) Comparison of the first principal component in the data and in a sample simulation as vectors.

Figure 20: Out-of-sample (four years after training) first principal component of the daily log implied volatility increments.



(a) Computed using SPX implied volatility data. (b) Computed using a sample VolGAN output.

Figure 21: Out-of-sample (four years after training) second principal component of the daily log implied volatility increments.



(a) Computed using SPX implied volatility data. (b) Computed using a sample VolGAN output.

Figure 22: Out-of-sample (four years after training) third principal component of the daily log implied volatility increments.

In order to quantify the similarity between the PCs of the simulated and the market data, we calculate the inner product between them (as vectors) over 1000 i.i.d. VOLGAN samples. A value of one would indicate perfect alignment of the eigenvectors. From Table 5 we note that the first two inner products (PC1 with PC1, and PC2 with PC2) are very close to one, especially considering that the

quantities are for the out-of-sample data. The inner product between the third eigenvectors of simulations and data realizations is lower than for the first two PCs, but it is nevertheless high. Furthermore, there is close resemblance in the physical interpretations of the third eigenvectors. Therefore, VOLGAN is able to learn the most important eigenvectors both qualitatively and quantitatively, showing the ability to learn the covariance structure of the SPX implied volatility co-movements.

Rank	Mean	Median	Standard deviation
First	0.921	0.922	0.009
Second	0.921	0.922	0.011
Third	0.798	0.798	0.011

Table 5: Out-of-sample inner products of eigenvectors of the covariance matrices of daily log-returns of SPX implied volatility and the corresponding eigenvectors of the covariance matrix of VolGAN implied volatility increments.

4.2.6 Correlation structure of variables

We further investigate VOLGAN’s ability to simulate realistic scenarios by examining how well it reproduces correlations between variables of interest. First, we consider the relationship between the projections of the log-implied volatility increments onto the first three principal components and the log-returns of the underlying.

Table 6 considers the correlations between index returns and the projections of the log-implied volatility increments onto the first three principal components, comparing their values in SPX options data with those in VOLGAN scenarios. The correlation between the first projection process and the simulated log-returns of the underlying is close to that of market data, whereas the projections on the second and the third principal component have slightly stronger correlations with the returns of the underlying in VOLGAN than they do in the SPX implied volatility market data. Nevertheless, both quantities are on the same scale. The correlation between the projection on the third principal component and the underlying is low both in VOLGAN and in the options data. VOLGAN is able to reproduce the correct relationships between the projection processes and the returns of the underlying: the correlations between the returns of the underlying and the projections of the log implied volatility increments onto the level and skew principal component are negative, whereas the correlation with the projection onto the curvature principal component is low (and positive).

PC rank	Data (test)	VOLGAN (test)	Data (train)
First	-0.76	-0.84 ± 0.024	-0.34
Second	-0.29	-0.38 ± 0.055	-0.32
Third	0.06	0.16 ± 0.020	0.28

Table 6: Pearson correlation between (simulated) SPX log-returns and the projections of the (simulated) log-implied volatility increments on the principal components. The VOLGAN column contains the mean $\pm 1.96 \times$ standard deviation of the observed Pearson correlations across 1000 samples. Implied volatility increments in the *Data (train)* column are projected onto the principal components of the test data for consistency.

In order to correctly capture joint dynamics of implied volatilities and the underlying index, we are interested in the relationship between the log increments of the index ($\Delta \log S_t$), the projection of the log-implied volatility increments onto the first principal component (ΔX_t^1), the log increments of the 1-month at-the-money implied volatility ($\Delta \log \sigma_t^{ATM}$), and the log increments of VIX ($\Delta \log v_t$). Table 7 contains average Pearson correlations for VOLGAN simulations (blue) vs the market data (red) on the test set. VOLGAN simulations exhibit similar correlations between all variables of interest. The correlations between the VIX increments and the increments of the other state variables are slightly lower in VOLGAN scenarios compared to the data observation on the test set. However, they are of the correct sign and magnitude. The correlation between $\Delta \log S_t$ and $\Delta \log \sigma_t^{ATM}$ became significantly higher in magnitude in the period used for testing compared to the period used for training, as noted in [Cont and Vuletic, 2023], which could explain why VOLGAN results in slightly stronger correlations between the index returns and ΔX_t^1 , that is $\Delta \log \sigma_t^{ATM}$

	$\Delta \log S_t$	ΔX_t^1	$\Delta \log \sigma_t^{ATM}$	$\Delta \log v_t$
$\Delta \log S_t$	1.00	-0.84 -0.76	-0.86 -0.77	-0.55 -0.71
ΔX_t^1	-0.84 -0.76	1.00	0.95 0.89	0.66 0.84
$\Delta \log \sigma_t^{ATM}$	-0.86 -0.77	0.95 0.89	1.00	0.72 0.96
$\Delta \log v_t$	-0.55 -0.71	0.66 0.84	0.72 0.96	1.00

Table 7: Out-of-sample (4.5 years after training including Covid) average Pearson correlation for simulated vs real values of log-returns of SPX ($\Delta \log S_t$), implied volatility level factor (ΔX_t^1), 1-month ATM volatility ($\Delta \log \sigma_t^{ATM}$) and VIX ($\Delta \log v_t$). Average VOLGAN outcome (blue) and data (red).

We repeat the analysis for the first year in the test set in Table 8. We observe that the magnitude of the correlation between the log-increments of VIX and the log SPX returns is a bit lower in simulations compared to the data. In the last year of the test set (Feb 2022-Feb 2023), the correlations between the simulated values of log SPX returns, increments of the level factor, and the at-the-money vol returns increase in magnitude, as noted in Table 9. We observe that the same is true for the actual values stemming from the data. The correlation structure of the simulated variables is consistent with the market, regardless of the testing period.

	$\Delta \log S_t$	ΔX_t^1	$\Delta \log \sigma_t^{ATM}$	$\Delta \log v_t$
$\Delta \log S_t$	1.00	-0.67 -0.66	-0.73 -0.82	-0.34 -0.80
ΔX_t^1	-0.67 -0.66	1.00	0.89 0.75	0.64 0.74
$\Delta \log \sigma_t^{ATM}$	-0.73 -0.82	0.89 0.75	1.00	0.77 0.96
$\Delta \log v_t$	-0.34 -0.80	0.64 0.74	0.77 0.96	1.00

Table 8: First year out-of-sample average Pearson correlation for simulated vs real values of log-returns of SPX ($\Delta \log S_t$), implied volatility level factor (ΔX_t^1), 1-month ATM volatility ($\Delta \log \sigma_t^{ATM}$) and VIX ($\Delta \log v_t$). Average VOLGAN outcome (blue) and data (red).

	$\Delta \log S_t$	ΔX_t^1	$\Delta \log \sigma_t^{ATM}$	$\Delta \log v_t$
$\Delta \log S_t$	1.00	-0.94 -0.80	-0.92 -0.72	-0.63 -0.76
ΔX_t^1	-0.94 -0.80	1.00	0.97 0.96	0.71 0.95
$\Delta \log \sigma_t^{ATM}$	-0.92 -0.72	0.97 0.96	1.00	0.76 0.95
$\Delta \log v_t$	-0.63 -0.76	0.71 0.95	0.76 0.95	1.00

Table 9: Last year out-of-sample average Pearson correlation for simulated vs real values of log-returns of SPX ($\Delta \log S_t$), implied volatility level factor (ΔX_t^1), 1-month ATM volatility ($\Delta \log \sigma_t^{ATM}$) and VIX ($\Delta \log v_t$). Average VOLGAN outcome (blue) and data (red).

Our results demonstrate that VOLGAN is able to simulate realistic co-movements for implied volatilities across a range of moneyness and maturities,

as well as the underlying index and VIX: in particular we are able to reproduce time-varying correlations between increments of these variables.

5 Application to hedging and risk management of option portfolios

The main motivation for generative models in finance is their use for risk management and hedging. We will now examine how VOLGAN may be used to design effective hedging strategies for options portfolios. In contrast with model-based approaches such as Deep hedging [Buehler et al., 2019], our approach is completely *data-driven* and model-free, in the spirit of [Hutchinson et al., 1994].

Consider a portfolio whose value $V_t = V(t, S_t, \sigma_t)$ is determined by the price of the underlying asset S_t and the implied volatility surface $\sigma_t(.,.)$. This may be a portfolio of call/put options, or any portfolio of derivatives which may be priced by calibrating a pricing model to the market volatility surface σ_t .

As an example, we will focus below on the case where the target portfolio is composed of (possibly illiquid) call or put options options, with the same expiry date T .

A typical problem is to hedge this portfolio with other, more liquid call/put options and the underlying. Let \mathcal{H} be the set of hedging instruments. A hedging strategy will be a self-financing portfolio composed of instruments $i \in \mathcal{H}$. If ϕ_t^i is the position (hedge ratio) in a hedging instrument, the value of the hedging portfolio \hat{V} satisfies

$$\Delta\hat{V}_t = \hat{V}_{t+\Delta t} - \hat{V}_t = \sum_{i \in \mathcal{H}} \phi_t^i \Delta H_t^i + r_t \left(\hat{V}_t - \sum_{i \in \mathcal{H}} \phi_t^i H_t^i \right) \Delta t, \quad (22)$$

where Δt is the hedging frequency and r_t is the risk-free interest rate. Set

$$\hat{V}_0 = V_0. \quad (23)$$

and denote by Z_t the tracking error, which is also the PnL of the hedged position:

$$Z_t = V_t - \hat{V}_t. \quad (24)$$

We will now compare several methods for choosing the hedging instruments and hedge ratios.

Delta hedging Here the only hedging instrument is the underlying $H_t^0 = S_t$ and the hedge ratio is set to be the overall (Black-Scholes) delta of the portfolio:

$$\phi_t^0 = \Delta_t^V = \frac{\partial V}{\partial S}(t, S_t, \sigma_t) \quad (25)$$

Delta-vega hedging This method achieves vega-neutrality by computing a sensitivity κ_t^V to a shift in implied volatilities (e.g. a parallel shift) and hedging against this move by including an option in the hedging set. As before, let H_t^0 be the underlying and let H_t^1 be the option used as hedging instrument. Typically this is a liquid call or put option. Denoting by κ_t^H the vega of the option used as hedging portfolio, we achieve vega-neutrality by choosing

$$\phi_t^1 = \frac{\kappa_t^V}{\kappa_t^H}, \quad \phi_t^0 = \Delta_t^V - \phi_t^1 \Delta_t^H. \quad (26)$$

where Δ_t^H is the delta of the option used for hedging.

Scenario-based regression hedging We now explain how to use VOLGAN to design a completely data-driven hedging strategy.

Given a set of hedging instruments $H^i, i \in \mathcal{H}$ and a set of VOLGAN next-day scenarios $\{\omega_j, j = 1..N\}$, we determine the hedge ratios ϕ_t^i by interpreting the one-step evolution of the portfolio

$$V_{t+\Delta t} - V_t = \sum_{i \in \mathcal{H}} \phi_t^i (H_{t+\Delta t}^i - H_t^i) + (Z_t - Z_{t+\Delta t}),$$

as a regression equation across VOLGAN scenarios:

$$\Delta V_t = V_{t+\Delta t}(\omega_j) - V_t = A_t + \sum_{i \in \mathcal{H}} \phi_t^i (H^i(\omega_j)_{t+\Delta t} - H_t^i) + \epsilon_j. \quad (27)$$

Therefore, the hedge ratios ϕ_t^i can be obtained by regressing the simulated values of ΔV_t on the corresponding simulated values of $\{\Delta H_t^i\}$.

Choice of hedging instruments Delta-vega hedging rules provide no insight on the choice of the hedging instrument and can be achieved in principle using any option used as hedging instrument. It is common to use ATM calls, but vega is sensitive to moves in the underlying asset. Our regression approach allows to choose the hedging instruments from a larger set of potential candidates \mathcal{H}_0 using variable selection methods such as LASSO, which induces sparsity and stability.

5.1 Example: hedging a straddle

In order to test how well VOLGAN captures the joint dynamics of the implied volatility surface and the underlying index, we perform a hedging exercise where the portfolio consists of a one-month call and put option with strike $K = 1.2S_0$. We will compare the following:

- **BS delta-vega hedge:** Black-Scholes delta-vega hedge using a call option initiated ATM at $t=0$.
- **BS delta hedge:** Black-Scholes delta hedging.

- **VOLGAN + LASSO:** VolGAN daily regression hedge with multiple options selected via LASSO regression, without scenario re-weighting ($\beta = 0$).
- **VOLGAN + ATM:** VolGAN daily regression hedge using a call option initiated ATM at $t = 0$, without scenario re-weighting ($\beta = 0$).
- **VOLGAN + LASSO + Scenario Weighting:** VolGAN daily regression hedge using call initiated ATM at $t = 0$, with scenario re-weighting.

The extended hedging set \mathcal{H}_0 used for LASSO selection in data-driven hedging via VOLGAN consists of calls and puts with the same expiry as the straddle position (one month from the start) and strikes of:

- $0.9S_0, 0.95S_0, 0.975S_0$ for puts
- $S_0, 1.025S_0, 1.05S_0, 1.1S_0$ for calls.

The hedging exercise is performed over the entire test set, with no overlapping periods. That is, each long straddle position is hedged until expiry, after which a new straddle position is entered.

We use LASSO for the selection of hedging instruments \mathcal{H} . To calibrate the L_1 regularization parameter by examining the in-sample R^2 and the Mean Squared Error as a function of the penalization parameter for the day on which a new position is entered. We repeat the same procedure every time we enter a new straddle. The values of α under consideration are from 0 to 1 in 0.1 increments. LASSO regression is used for instrument selection at time $t = 0$ only. After the hedging instruments have been selected, the hedge ratios are computed via ordinary least squares.

We first explore whether scenario re-weighting improves the hedging performance or not. That is, we compare the two LASSO-based methods, using the same values of α (chosen using raw outputs). Both methods perform regression using 1000 samples from VOLGAN. The tracking error histogram in Figure 23 shows that it is better to use raw VOLGAN outputs, since they mimic the market, rather than applying arbitrage-scenario re-weighting. In the remainder of this section, we will focus on raw VOLGAN outputs.

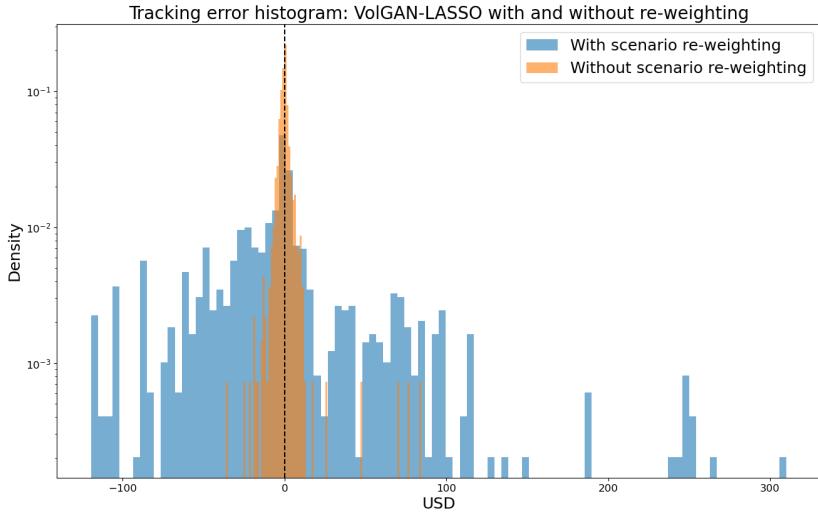


Figure 23: Distribution of tracking error: data-driven hedging via VOLGAN and LASSO regression, with and without scenario re-weighting.

Figure 25 shows that the algorithm typically picks 2 or 3 options as hedging instruments during periods of market turbulence, which shows that the portfolio dynamics is well represented by a 2- or 3-factor (implied) volatility process. This result is consistent with the principal component analysis results for VolGAN outputs (Table 4), which show 3 significant factors driving the implied volatility co-movements [Cont and da Fonseca, 2002, Cont and Vuletic, 2023].

There are periods during which no options are selected for hedging, in line with the straddle vega being zero (Figure 24a). Up until the start of the Covid-19 pandemic the straddle delta is equal to minus one (Figure 24b), and the straddle vega is zero. The instances with no regularization correspond to 7 options used for hedging.

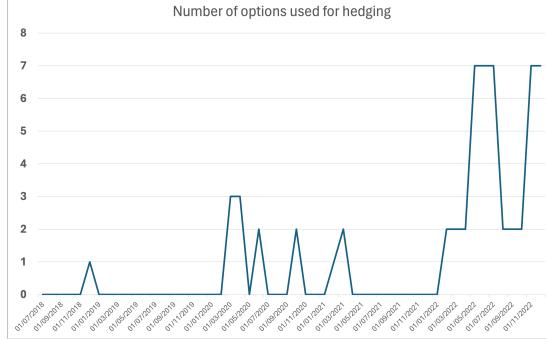


Figure 25: Number of hedging instruments selected using LASSO across VolGAN scenarios. During periods of calm no options are used for hedging. During periods of market turbulence, usually 2-3 options are selected for hedging, indicating that portfolio dynamics is well represented by a 2- or 3-factor (implied) volatility process.

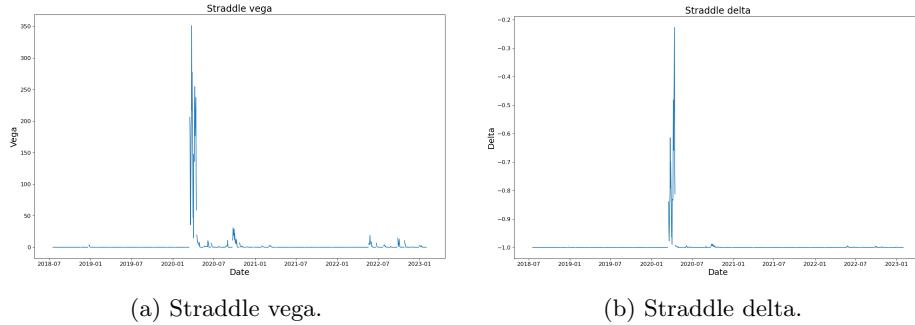


Figure 24: Black-Scholes vega (κ_t^V) and delta (Δ_t^V) of the straddle portfolio on the test set. We note a jump in both values at the start of the Covid-19 pandemic.

Figure 25 shows that, during the Covid-19 pandemic and the start of Ukraine war, 2-3 options are used for hedging.

Table 10 offers a summary of how many times each option is used for hedging. In all but one instance in which options are used, the call initiated at-the-money was included. The remaining call options were only used in the 5 periods during which no regularization was applied (when $\alpha = 0$ due to our search grid). When a single option was used, it was the at-the-money call.

These examples illustrate that VOLGAN is more flexible than a factor model with a fixed number of factors: the number of effective factors, which corresponds to the number of hedging instruments used, changes dynamically with market conditions.

Option Type	Initial moneyness	Frequency
Put	0.9	6
Put	0.95	7
Put	0.975	16
Call	1	16
Call	1.025	5
Call	1.05	5
Call	1.1	5

Table 10: Frequency of options selected by LASSO in Methods 3 (no re-weighting) and 5 (with re-weighting).

Before comparing VOLGAN-driven methods with delta hedging and delta-vega hedging, we discuss the differences between using the automatic instrument selection via LASSO regression and the call initiated at-the-money. Figure 26 shows the tracking error histograms produced by the two methods. Automatic selection of hedging instruments appears to result in a more concentrated tracking error distribution around zero.

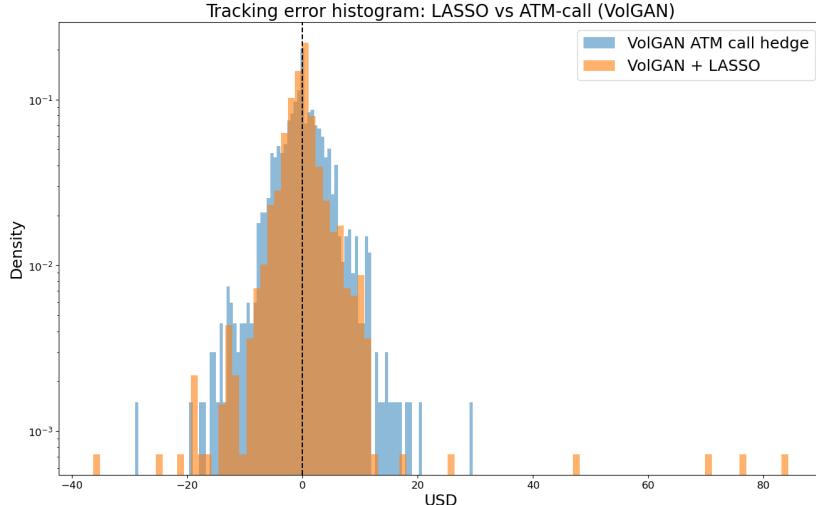


Figure 26: Tracking error distribution: VOLGAN + LASSO vs VOLGAN at-the-money call hedge.

We compare the two data-driven approaches using raw VOLGAN outputs with delta hedging and delta-vega hedging. Figure 27 indicates that away from the initial Covid-19 shock all methods replicate the straddle well. However, during this period, delta-vega hedging shows instability and results in a significant decrease in the hedging portfolio value \hat{V}_t . This does not happen with the other methods, including the VOLGAN at-the-money hedge, despite the fact that the

two use exactly the same hedging instruments. Such behavior highlights the fact that the delta-vega hedge ratio corresponding to the option might become unstable, especially if the vega of the option used for hedging becomes small. The corresponding tracking errors Z_t as functions of time are displayed in Figure 28.

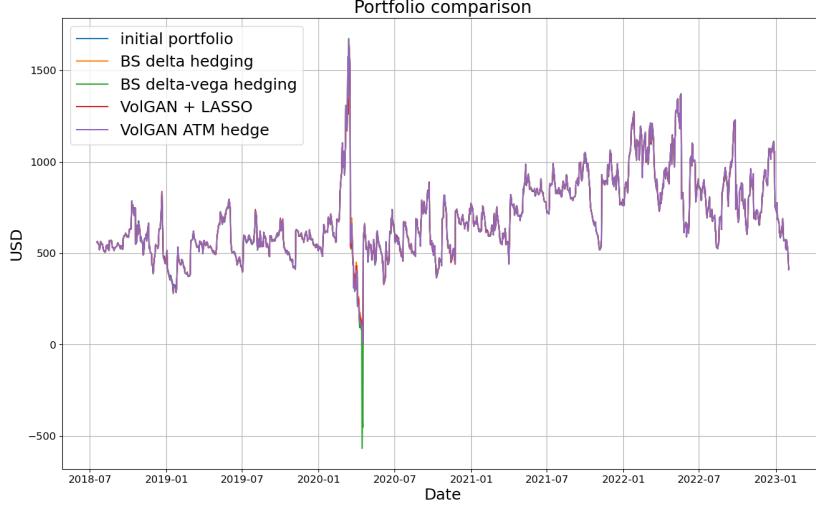


Figure 27: Straddle V_t and hedging portfolios \hat{V}_t corresponding to different methods. Each new straddle position is entered the day after expiry of the previous one.

Table 11 contains the tracking error statistics such as mean, standard deviation, and Value-at-Risk (5%, 2.5%, 1%) for the Black-Scholes and VOLGAN hedging methods. Value-at-Risk (VaR) at level a is calculated as the negative a quantile of the tracking error. VOLGAN + LASSO result in the mean closest to zero, the second lowest standard deviation, and in the lowest Value-at-Risk metrics. The lowest standard deviation is achieved by the ATM hedging via VOLGAN. However, comparing the histograms in Figure 26 and the tracking errors in Figure 28, we note that the increase in standard deviation when opting for automatic hedging instrument selection is due to the performance during April 2020. The bulk of the tracking error distribution is slightly tighter for VOLGAN + LASSO than for VOLGAN + ATM. Delta-vega hedging has lower VaR than delta-hedging, but higher standard deviation, due to the instability evidenced in April 2020. All tracking error distributions are compared in Figure 29. In this instance, data-driven hedging via VOLGAN outperforms the Black-Scholes benchmarks. This is particularly impressive given the volatility and the length of the testing period. This test shows that VOLGAN is indeed able to capture the co-movements of the implied volatility surface and the underlying.

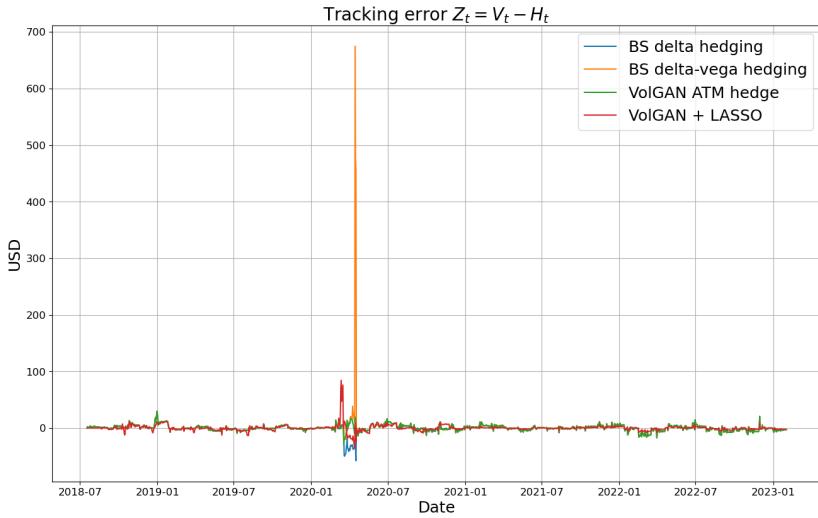


Figure 28: Tracking error Z_t corresponding to different methods. Each new straddle position is entered the day after expiry of the previous one.

Statistics	VolGAN + LASSO	VolGAN + ATM	BS delta	BS delta-vega
Mean	-0.051	0.056	-0.614	1.541
St.dev	5.766	4.940	6.755	28.307
5% VaR	5.815	7.314	8.310	7.258
2.5% VaR	8.095	10.692	13.300	10.701
1% VaR	13.172	13.730	34.023	14.181

Table 11: Tracking error Z_t statistics (mean, standard deviation, 5% Value-at-Risk, 2.5% Value-at-Risk, and 1% Value-at-Risk) as obtained by different models (in USD).

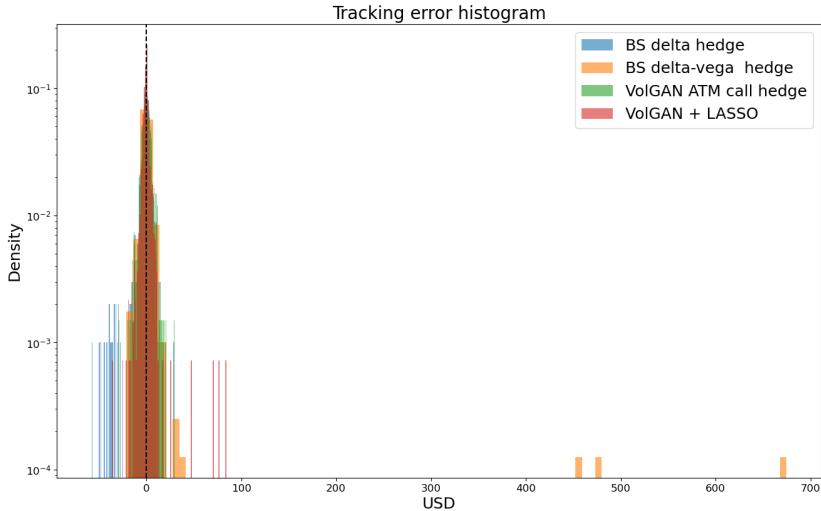


Figure 29: Histogram of the tracking error Z_t corresponding to different methods. Each new straddle position is entered the day after expiry of the previous one.

Code availability

VOLGAN code is available on GitHub: <https://github.com/milenavuletic/VolGAN/>.

Data availability

SPX options data is available from OptionMetrics. VIX times series is available from CBOE (www.cboe.com/).

Funding

Milena Vuletić's research was supported by BNP PARIBAS through the *EPSRC Centre for Doctoral Training in Mathematics of Random Systems: Analysis, Modelling and Simulation* (ESPRC Grant EP/S023925/1).

Acknowledgements

We thank Katia Babbar, Andrey Chirikhin, Samuel N. Cohen, Mihai Cucuringu, Bruno Dupire, Blanka Horvath, Botao Li, Terry Lyons, Fabio Mercurio, Christoph Reisinger, Jan Obloj, Justin Sirignano and seminar participants at BNP Paribas PhD Days 2023, QuantMinds 2023 and Quant Summit Europe 2023 for helpful comments and remarks.

References

- [Avellaneda et al., 2020] Avellaneda, M., Healy, B., Papanicolaou, A., and Papanicolaou, G. (2020). PCA for Implied Volatility Surfaces. *The Journal of Financial Data Science*, 2(2):85–109.
- [Babbar, 2001] Babbar, K. A. (2001). *Aspects of stochastic implied volatility in financial markets*. PhD thesis, Imperial College London.
- [Bonett and Wright, 2000] Bonett, D. G. and Wright, T. A. (2000). Sample size requirements for estimating Pearson, Kendall and Spearman correlations. *Psychometrika*, 65:23–28.
- [Buehler et al., 2019] Buehler, H., Gonon, L., Teichmann, J., and Wood, B. (2019). Deep hedging. *Quantitative Finance*, 19(8):1271–1291.
- [Cao et al., 2020] Cao, J., Chen, J., and Hull, J. (2020). A neural network approach to understanding implied volatility movements. *Quantitative Finance*, 20(9):1405–1413.
- [CBOE, 2022] CBOE (2022). Volatility Index Methodology: Cboe Volatility Index. https://cdn.cboe.com/api/global/us_indices/governance/VIX_Methodology.pdf. [Online; accessed 8-May-2023].
- [Cohen et al., 2022] Cohen, S., Reisinger, C., and Wang, S. (2022). Estimating risks of european option books using neural-SDE market models. *Journal of Computational Finance*, 6(3):33–72.
- [Cont et al., 2022] Cont, R., Cucuringu, M., Xu, R., and Zhang, C. (2022). Tail-GAN: Nonparametric Scenario Generation for Tail Risk Estimation. *arXiv preprint arXiv:2203.01664*.
- [Cont and da Fonseca, 2002] Cont, R. and da Fonseca, J. (2002). Dynamics of implied volatility surfaces. *Quantitative Finance*, 2(1):45–60.
- [Cont et al., 2002] Cont, R., Fonseca, J. d., and Durleman, V. (2002). Stochastic models of implied volatility surfaces. *Economic Notes*, 31(2):361–377.
- [Cont and Tankov, 2004] Cont, R. and Tankov, P. (2004). *Financial modelling with jump processes*. CRC Press.
- [Cont and Vuletic, 2023] Cont, R. and Vuletic, M. (2023). Simulation of arbitrage-free implied volatility surfaces. *Applied Mathematical Finance*, 30:94–121.
- [Cuchiero et al., 2020] Cuchiero, C., Khosrawi, W., and Teichmann, J. (2020). A generative adversarial network approach to calibration of local stochastic volatility models. *Risks*, 8(4):101.
- [Davis and Hobson, 2007] Davis, M. H. and Hobson, D. G. (2007). The range of traded option prices. *Mathematical Finance*, 17(1):1–14.

- [Dumas et al., 1998] Dumas, B., Fleming, J., and Whaley, R. E. (1998). Implied volatility functions: Empirical tests. *The Journal of Finance*, 53(6):2059–2106.
- [Dupire, 1994] Dupire, B. (1994). Pricing with a smile. *Risk*, 7(1):18–20.
- [Durrleman, 2010] Durrleman, V. (2010). From implied to spot volatilities. *Finance and Stochastics*, 14:157–177.
- [Gatheral, 2011] Gatheral, J. (2011). *The volatility surface: a practitioner’s guide*. John Wiley & Sons.
- [Gerhold and Güüm, 2020] Gerhold, S. and Güüm, I. C. (2020). Consistency of option prices under bid–ask spreads. *Mathematical Finance*, 30(2):377–402.
- [Goodfellow et al., 2014] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.
- [Heston, 1993] Heston, S. L. (1993). A closed-form solution for options with stochastic volatility with applications to bond and currency options. *The review of financial studies*, 6(2):327–343.
- [Hinton et al., 2012] Hinton, G., Srivastava, N., and Swersky, K. (2012). Neural Networks for Machine Learning, Lecture 6. Coursera.
- [Hutchinson et al., 1994] Hutchinson, J. M., Lo, A. W., and Poggio, T. (1994). A nonparametric approach to pricing and hedging derivative securities via learning networks. *Journal of Finance*, 49(3):851–889.
- [Jackson et al., 1999] Jackson, N., Suli, E., and Howison, S. (1999). Computation of deterministic volatility surfaces. *Journal of Computational Finance*, 2(2):5–32.
- [Kingma et al., 2019] Kingma, D. P., Welling, M., et al. (2019). An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4):307–392.
- [Mirza and Osindero, 2014] Mirza, M. and Osindero, S. (2014). Conditional Generative Adversarial Nets.
- [Ning et al., 2023] Ning, B. X., Jaimungal, S., Zhang, X., and Bergeron, M. (2023). Arbitrage-free implied volatility surface generation with variational autoencoders. *SIAM Journal on Financial Mathematics*, 14(4):1004–1027.
- [OptionMetrics, 2021] OptionMetrics (2021.). IvyDB US Reference Manual. https://wrds-www.wharton.upenn.edu/documents/1504/IvyDB_US_Reference_Manual_rn2hAXz.pdf. Version 5.0.

- [Sana and Cont, 2005] Sana, B. H. and Cont, R. (2005). Recovering volatility from option prices by evolutionary optimization. *Journal of Computational Finance*, 8(4):43–76.
- [Schönbucher, 1999] Schönbucher, P. J. (1999). A market model for stochastic implied volatility. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 357(1758):2071–2092.
- [Takahashi et al., 2019] Takahashi, S., Chen, Y., and Tanaka-Ishii, K. (2019). Modeling financial time-series with generative adversarial networks. *Physica A: Statistical Mechanics and its Applications*, 527:121261.
- [Vuletić et al., 2023] Vuletić, M., Prenzel, F., and Cucuringu, M. (2023). Fin-gan: Forecasting and classifying financial time series via generative adversarial networks. *Quantitative Finance*, to appear.
- [Wiese et al., 2019] Wiese, M., Bai, L., Wood, B., and Buehler, H. (2019). Deep hedging: learning to simulate equity option markets. *arXiv preprint arXiv:1911.01700*.
- [Wiese et al., 2020] Wiese, M., Knobloch, R., Korn, R., and Kretschmer, P. (2020). Quant GANs: deep generation of financial time series. *Quantitative Finance*, 20(9):1419–1440.
- [Wiese et al., 2021] Wiese, M., Wood, B., Pachoud, A., Korn, R., Buehler, H., Murray, P., and Bai, L. (2021). Multi-Asset Spot and Option Market Simulation.