Neural Tangent Kernel in Implied Volatility Forecasting: A Nonlinear Functional Autoregression Approach

Ying Chen[*] Maria Grith[†] Hannah L. H. Lai[‡]

Abstract

Forecasting implied volatility across different levels of moneyness and maturity is crucial yet challenging due to the high dimensionality of the Implied Volatility Surface (IVS) and the nonlinearity that characterizes its temporal dependence. We adopt a Nonlinear Functional Autoregressive (NFAR) framework to a sequence of IVS and employ neural networks that admit a Neural Tangent Kernel (NTK) parametrization to capture nonlinear interactions between surfaces. We illustrate the theoretical and numerical advantages of the proposed functional NTK (fNTK) estimator and establish a link to functional kernel regression. Our empirical analysis includes over 6 million European calls and put options from the S&P 500 Index, covering January 2009 to December 2021. The results confirm the superior forecasting accuracy of the fNTK across different time horizons. When applied to short delta-neutral straddle trading, the fNTK achieves a Sharpe ratio ranging from 1.30 to 1.83 on a weekly to monthly basis, translating to 90% to 675% relative improvement in portfolio returns compared to forecasts based on functional Random Walk model.

Keywords: Implied Volatility Surface; Neural Networks; Neural Tangent Kernel; Implied Volatility Forecasting; Nonlinear Functional Autoregression; Option Trading Strategies.

JEL classification: C14, C45, C58, G11, G13, G17

[*] E-mail: matcheny@nus.edu.sg. Department of Mathematics, Asian Institute of Digital Finance, Risk Management Institute, National University of Singapore

[†] E-mail: grith@ese.eur.nl. Erasmus School of Economics, Erasmus University Rotterdam

[‡] E-mail: hlhlai@u.nus.edu. Department of Mathematics, Integrative Sciences and Engineering Programme, National University of Singapore

1 Introduction

Options implied volatility (IV) is a key metric of financial market expectations and sentiment. IV is instrumental in analyzing option pricing trends and developing trading strategies. Given its role in financial decision-making, accurate IV forecasting is essential, highlighting the need for innovative modeling techniques to maximize its potential in shaping informed financial strategies.

Adequate forecasting models of IV must account for its nonlinear dynamics (Audrino and Colangelo, 2010, Bloch and Böök, 2020, Almeida et al., 2022, Zhang, Li, and Zhang, 2023). They must also capture complex spatial dependence between IVs with different strikes and maturities by modeling them as observations of a smooth Implied Volatility Surface (IVS) whose shape effectively reflects market conditions (Aït-Sahalia and Lo, 1998, Cont and Da Fonseca, 2002, Fengler, Härdle, and Mammen, 2007, Park et al., 2009, Fengler and Hin, 2015). Modeling the spatial and temporal dependencies in the IVS series jointly without imposing restrictive assumptions is theoretically and computationally challenging due to the curse of dimensionality, as the number of model parameters far exceeds the number of samples. Simpler models may address some nonlinear dependencies but often cannot fully capture the intricate dynamics of IVS data.

In this article, we adopt a Nonlinear Functional Autoregressive (NFAR) framework to a sequence of IVS and employ neural networks to capture nonlinear interactions between surfaces. Our estimation approach, applied to IVS defined over an infinite domain of moneyness and maturity, utilizes sieves (Chen, 2007) alongside series expansion (Cho et al., 2013) and Neural Tangent Kernel (NTK) parametrized neural networks (Jacot, Gabriel, and Hongler, 2018, Arora et al., 2019 and Domingos, 2020). This methodology effectively overcomes the high dimensionality challenge associated with nonparametric modeling of spatial-temporal dependencies of IVS. We demonstrate the theoretical and numerical advantages of the proposed functional NTK (fNTK) estimator and establish a link to functional kernel regression.

Our work makes several key contributions that advance the IVS forecasting literature. First, it models the temporal dependence of IVS, adopting a fully nonparametric framework. The novelty of our approach is employing neural networks that admit an NTK parametrization to capture nonlinear temporal relations between curves. Specifically, we formulate the NFAR model by projecting the IVS onto orthogonal bivariate B-spline basis functions (Redd, 2012), while the NTK parameterizes the temporal dependence among coefficients. By incorporating the NTK within a functional data setup, the fNTK estimator leverages the information in the entire IVS and facilitates the approximation of complex nonlinear dependence with deep neural networks.

Second, we illustrate the connection between the fNTK estimator and functional kernel regression. A kernel facilitates the transition from a nonlinear to a linear regression problem, wherein the kernel utilized plays a straightforward role in capturing nonlinear dependencies. Using a kernel representation in our work enables the transitions from function to vector-based analysis. Namely, we establish an isomorphism between vector and function kernel spaces and show the consistency of the fNTK estimator. In this way, we enhance traditional functional kernel regressions with established kernels (e.g., Kadri et al., 2010, Li and Song, 2017, and Sang and Li, 2022) to models that utilize more general neural kernels, which represent an innovation to the nonlinear function-on-function regression literature.

Through extensive empirical analysis of over 6 million European options from the S&P 500 Index, our fNTK estimator demonstrates superior forecasting accuracy and trading performance, including a significant reduction in prediction errors and enhanced Sharpe ratios in trading strategies compared to several benchmark models. Specifically, the fNTK estimator shows a notable improvement in forecasting accuracy, with 29% to 45% reduction in the RMSE for predictions compared to functional Random Walk (fRW) benchmark on a weekly to monthly basis. Additionally, we explore the economic value of accurate IV forecasting for different trading strategies. For instance, using fNTK in short delta-neutral straddle trading leads to Sharpe ratios between 1.30 and 1.83, translating to 90% to 675% relative enhancement in portfolio returns compared to an alternative based on fRW. Our analysis highlights the competitive performance of fNTK-supported strategies during highly volatile periods and for longer investment horizons. These results underscore the practical effectiveness of our modeling approach.

The paper is structured as follows. Next, we review related literature. Section 2 comprehensively describes the data. Section 3 introduces the NFAR model, explains the estimation procedure via fNTK, and discusses the consistency of our estimator within a functional kernel regression. In Section 4, we report on the model's implementation using data. Section 5 examines the economic value of fNTK forecasts. The paper concludes with Section 6.

1.1 Related Literature

The article primarily focuses on the literature on modeling implied volatility surfaces and their dynamics, with various methods proposed to account for the intricate spatial and temporal dependencies. The existing literature often uses parametric models to estimate or calibrate the IVS.[1] However, these parametric models rely on predetermined features to represent the IVS and can only capture limited information. In contrast, nonparametric modeling of the IVS takes full advantage of the available information from the observed IVs. These include classical kernel smoothing methods (Aït-Sahalia and Lo, 1998, Cont and Da Fonseca, 2002) and basis expansion (Fengler, Härdle, and Mammen, 2007, Park et al., 2009, Fengler and Hin, 2015), and also modern machine learning techniques such as tree models (Audrino and Colangelo, 2010), neural networks (Ackerer, Tagasovska, and Vatter, 2020, Almeida et al., 2022), autoencoders (Bergeron et al., 2022), generative adversarial networks (Vuletić and Cont, 2023).

Concurrently, numerous studies have analyzed the dynamics of IVS, with most of the early studies focusing on linear models.[2] More recent models integrate machine learning techniques to capture nonlinear temporal dependencies, such as Audrino and Colangelo (2010), Bloch and Böök (2020), Almeida et al. (2022) and Zhang, Li, and Zhang (2023). Our work is mainly related

[1] Classical parametric models include the Black-Scholes model (Black and Scholes, 1973), the Heston model (Heston, 1993), the ad-hoc Black–Scholes model (Dumas, Fleming, and Whaley, 1998) and the Vega-Gamma-Vanna-Volga model (Carr and Wu, 2016). However, the trend in option pricing is to recognize the importance of incorporating information from the IVS. For instance, Aït-Sahalia, Li, and Li (2021a) and Aït-Sahalia, Li, and Li (2021b) propose implied stochastic volatility models whose coefficient functions reproduce key empirical features of observed IVS.

[2] To represent the dynamics, these models utilize parametric features (Goncalves and Guidolin, 2006, Bernales and Guidolin, 2014, and Bernales and Guidolin, 2015), principal components (Fengler, Härdle, and Villa, 2003, Cont and Da Fonseca, 2002), or factor models (Fengler, Härdle, and Mammen, 2007, Park et al., 2009, Fengler and Hin, 2015, and Ulrich and Walther, 2020).

to the last two articles. While all three studies, including ours, share a common objective of forecasting implied volatility surfaces and employing neural network methods to capture their complex, nonlinear dependencies over time, there are significant differences between our work and these previous articles in terms of methodology, scope, and implementation.

Unlike the approaches that restrict the IVS to low-dimensional or parametric forms, our work treats the IVS in its entirety, allowing for greater modeling flexibility. Specifically, in Zhang, Li, and Zhang (2023), the authors interpolate discrete IV points using the ad-hoc Black-Scholes (AHBS) model proposed by Dumas, Fleming, and Whaley (1998). They then apply dimensionality reduction techniques like Principal Component Analysis (PCA) or autoencoders to extract low-dimensional representations of the IV surface. In contrast, our methodology avoids imposing any predefined parametric structure on the IVS. Instead, we leverage orthogonal basis projection and the sieve method as described by Chen (2007), which enables the extraction of coefficients that fully describe the IV surface in a nonparametric manner.

While Almeida et al. (2022) aims to predict forecast errors from parametric models, our approach directly forecasts the entire implied volatility surface, providing more comprehensive insights into IV dynamics. Specifically, in Almeida et al. (2022), the authors primarily focus on predicting forecast errors from various parametric models such as the Black-Scholes model (Black and Scholes, 1973), the Heston model (Heston, 1993), the ad-hoc Black–Scholes model (Dumas, Fleming, and Whaley, 1998) and the Vega-Gamma-Vanna-Volga model (Carr and Wu, 2016). Their approach utilizes a feedforward neural network to correct the errors from these parametric models using the moneyness and time-to-maturity of the options contracts as input. In the options panel exercise, apart from observed features, their model incorporates time-varying covariates to capture the dynamic fluctuations of the IVs, a method resembling the approach of Audrino and Colangelo (2010). In contrast, our work directly forecasts the entire IVS rather than just correcting forecast errors and forecasts implied volatility surfaces based on lagged surfaces summarized by spline coefficients. By focusing on the comprehensive forecasting of the IVS, we aim to capture the entire temporal and cross-sectional dynamics of the volatility surface. This allows for a more holistic understanding of market conditions and potential trading opportunities rather than being restricted to refining existing parametric models.

The use of NTK parametrization in our work distinguishes our method from previous studies that employ neural networks in the context of forecasting implied volatility. Zhang, Li, and Zhang (2023) use Long Short-Term Memory (LSTM) neural networks to model the temporal dependence among the latent features of the IVS, while Almeida et al. (2022) use a feedforward neural network. In contrast, our approach utilizes the NTK parametrization within neural networks, which allows the model to approximate complex nonlinear dependencies effectively while benefiting from representation as kernel regression. This connection to kernel methods has been demonstrated both theoretically and empirically for the multivariate case in previous studies, such as Jacot, Gabriel, and Hongler (2018), Arora et al. (2019), and Lee et al. (2019). Our article extends the existing work by integrating NTK-parametrized neural networks into the functional kernel regression setting. Moreover, the numerical performance of the fNTK estimator is evidenced in the simulation and empirical studies.

Our approach also connects to the literature on nonlinear functional regression. Estimating nonlinear functional regression through classical nonparametric methods, such as kernel smoothing regression, suffers from slow convergence when the dimension of the regressors increases. Several approaches have been proposed in the functional data literature to tackle this challenge, typically by making additional assumptions on the underlying nonlinear dependence structure, such as functional additive models (Muller, Wu, and Yao, 2013; Müller and Yao, 2008), functional quadratic models (Sun and Wang, 2020) and functional index models (Chen, Hall, and Müller, 2011, Jiang and Wang, 2011). More recently, nonlinear function-on-function regression has been formulated as functional kernel regression in a Reproducing Kernel Hilbert Space (RKHS). We relate to this stream of literature and enhance conventional functional kernel regressions (e.g., those by Kadri et al., 2010, Li and Song, 2017, and Sang and Li, 2022) to more general neural network kernels like NTK, providing a more flexible class of functional models. Compared to the common parametrizations of the reproducing kernels, such as Gaussian and Laplacian, the NTK extracts features and learns the dependence jointly from the input and output, akin to an adaptive kernel. The introduction of NTK improves the model's predictive capability and supports further theoretical development in functional time series.

2 Data

We consider daily options on the S&P 500 Index obtained from IvyDB OptionMetrics for the period spanning from January 1, 2009, through to December 31, 2021, encompassing about 6.4 million European calls and puts. This repository facilitates a comprehensive insight into option contract specifics: best bid and ask quotes, expiration date, strike, implied volatility,[3] open interest and volume. We also collect data on the closing value of the S&P 500 index, dividend, forward prices, and the yield curve of risk-free interest rate proxy for constructing implied volatility surfaces and further analysis. The zero-coupon interest rates curve used by IvyDB is derived from ICE IBA LIBOR rates and settlement prices of CME Eurodollar futures.

In the field of implied volatility forecasting and trading strategies, SPX options are particularly valuable due to several factors. Firstly, they represent the most significant U.S.-based publicly traded companies, serving as a robust barometer for broad market volatility expectations. Their substantial liquidity, being among the most frequently traded options globally, ensures that the volatility signals derived are minimally affected by liquidity-induced noise. Furthermore, SPX options offer diverse expiration cycles and a wide range of strike prices, allowing for a detailed analysis of the implied volatility surfaces. Their cash-settled nature eliminates directional biases and the complexities associated with physical settlements. Lastly, the aggregated nature of SPX provides a more stable and consistent measure of market volatility compared to individual stocks.

Drawing inspiration from the methodology outlined by Büchner and Kelly (2022), we meticulously apply a set of filters to our dataset to eliminate entries that may arise from recording

[3] IvyDB computes the Black-Scholes implied volatility for a given option from the midpoint of the best closing bid price and best ask. The interest rate corresponds to the zero-coupon rate with a maturity equal to the option's expiration. It is obtained by linearly interpolating the two closest zero-coupon rates on the zero curve. The daily dividend rate is assumed constant and is computed by IvyDB from the put-call parity under a "constant dividend yield" assumption using the call's bid price with the offer price of the put and vice versa.

discrepancies or erroneous inputs. Specifically, we discard all options in which i) the bid price is negative or zero, ii) the bid exceeds the ask, iii) no-arbitrage conditions are violated,[4] iv) implied volatility is missing or non-positive, and (v) the open interest is negative.

For each option contract, we define time-to-maturity τ measured in years as the number of trading days to expiration divided by 252. To measure the moneyness of a contract, we follow Andersen, Fusari, and Todorov (2017) and define it as: $m = \frac{\ln(K/F_\tau)}{\sqrt{\tau} IV_{atm,\tau}}$, where K is the strike price, F_τ denotes the forward price for transactions τ years into the future, while $IV_{atm,\tau}$ denotes the annualized implied volatility of the option with the strike price closest to F_τ.[5] This definition of moneyness has two advantages: (i) an exactly at-the-money option (i.e., $K = F_\tau$) attains a delta of roughly 0.5 in absolute value, and (ii) it ensures that the implied volatility surfaces are comparable across different maturities and volatility regimes.

Our study models the implied volatility surfaces. The raw data are discretely observed at irregular points on the moneyness and maturity grid. We perform a pre-smoothing to translate the discrete observations on the same day into a daily continuous function. Compared to discrete implied volatilities, IVS holistically offers a comprehensive perspective, capturing the intricate dynamics across different strike prices and expiration times. By encapsulating rich volatility information across the entire options chain in a unified view, the inherent spatial relationships between implied volatilities at varying maturities and moneyness are seamlessly integrated. Even for SPX, specific moneyness and maturities might be sparse. Thus, we construct IV surfaces with tenors between 5 and 252 trading days, with moneyness ranging from $-2 \leq m \leq 2$. To construct smooth implied volatility surfaces using daily implied volatilities, we encompass options with moneyness stretching from -2.5 to a cap of 2.5 and a time-to-maturity τ of up to 280 days. Our consolidated dataset spans 3273 days, averaging 908 call options and 875 put options daily.[6]

The implied volatilities of put and call options with the same strike and maturity provided by OptionMetrics deviate from one another, thereby infringing upon the principles of put-call parity.

[4] For example, we ensure the monotonicity of option prices with respect to the strike price.

[5] If a forward contract for a desired time-to-maturity τ is not available, we apply linear interpolation between the two closest forward prices.

[6] In Appendix, Table A.1 summarizes the composition of our consolidated sample.

To address this issue, we construct the implied volatility surfaces of call and put options separately. By independently tailoring the IVS for puts and calls, we accommodate each option type's disparate dynamics and trading patterns. In Figure 1, we provide an illustration that showcases four IV surfaces for both call and put options: one from a calm period in 2019 and another from a volatile period in 2020. This figure also includes snapshots of the discrete implied volatilities of traded options for these periods. An IV surface clearly provides a smoother and more consistent representation of volatility across diverse moneyness and maturity combinations.

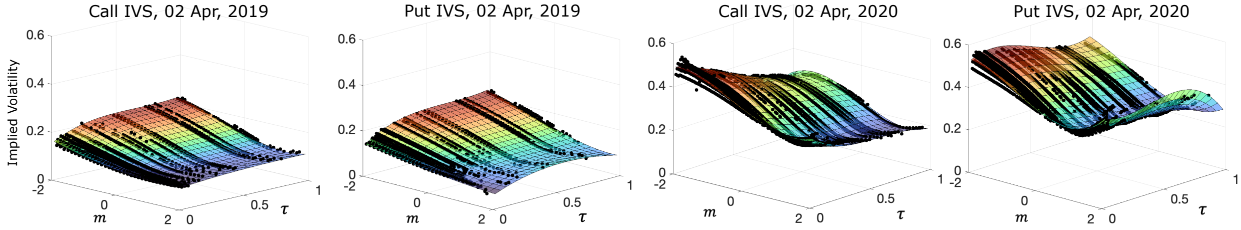
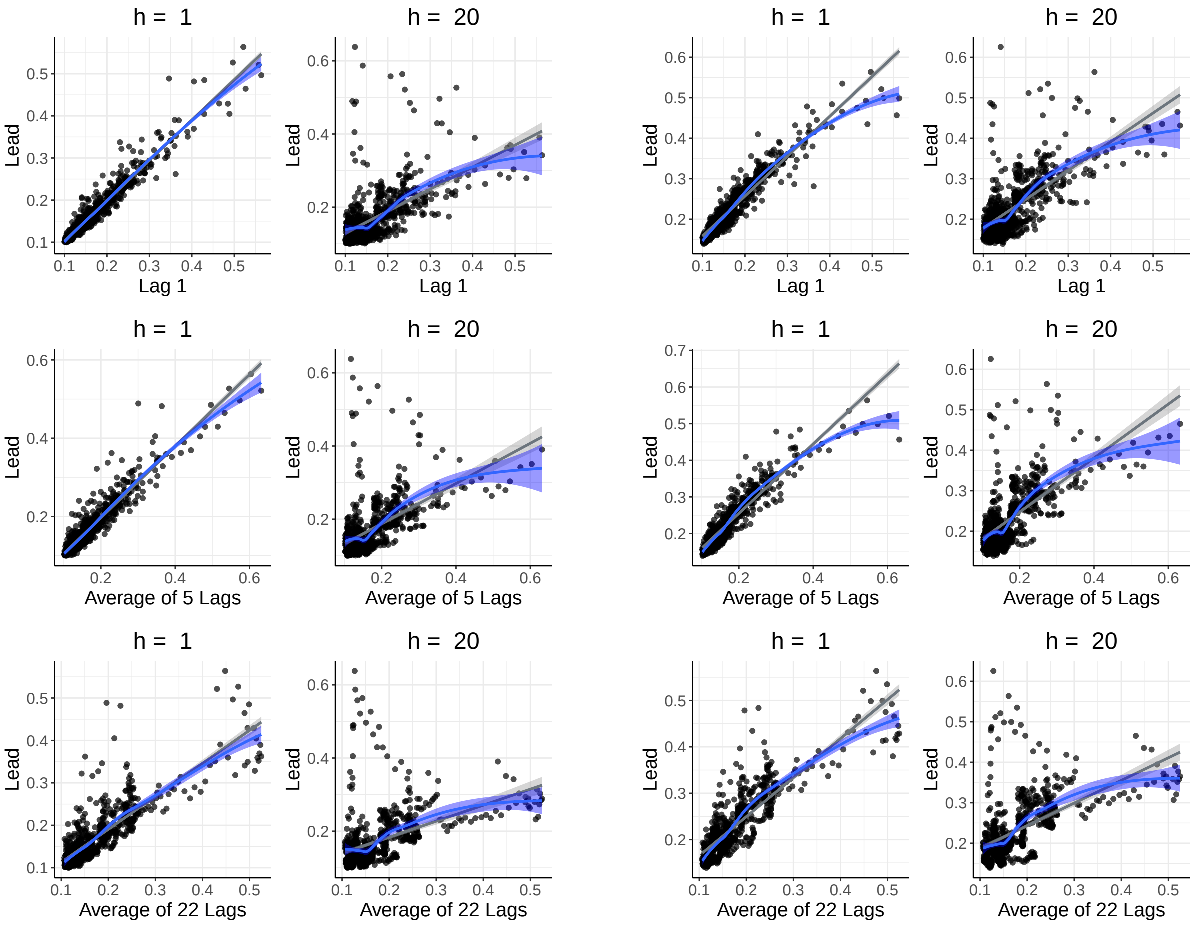


Figure 1: IVS of S&P500 calls and puts options on two different days with low and high volatility. The observed IV values are black dots, and the surfaces are smoothed with two-dimensional B-splines.

Figure 2 illustrates the lead-lag relationships and cross-lead-lag regressions for call options averages, highlighting the complex interconnectivity of implied volatility (IV) values across different days and maturities. This relationship exhibits nonlinear characteristics, as evidenced by the nonparametric fit shown in the figure, indicating nonlinear dynamics in IV, particularly over extended forecasting horizons. These intricate temporal dynamics suggest that while autoregressive models are necessary, the observed nonlinearities necessitate a more sophisticated nonlinear modeling approach.

3 Methodology

In this section, we introduce the nonlinear functional autoregressive (NFAR) model for implied volatility surfaces. This model operates within a nonlinear function-on-function regression framework, utilizing a kernel trick to linearize the nonlinear dependencies within the Reproducing Kernel Hilbert Space (RKHS). Our model is compatible with various kernels, and we specifically employ



(a) Lead-lag regression

(b) Cross lead-lag regression

Figure 2: Temporal dependence in the average IVS of call and put options. We plot the lead-lag (both IV on day i and on day $i+h$ at 30 day-to-maturity and moneyness $m=0$) as well as the cross-lead-lag regression (IV on day i at 30 day-to-maturity and $m=0$ while IV on day $i+h$ at 60 day-to-maturity and $m=-0.75$) at two horizons, $h=1$ and 20 and different lags (the last one day, average of the last five days, and average of the last 22 days), using data of the prediction period, from Jan 05, 2019 to Dec 31, 2021.

the Neural Tangent Kernel (NTK), a sophisticated kernel from advanced machine learning that effectively captures complex nonlinear relationships in the feature space through neural networks.

3.1 Nonlinear Functional Regression

We denote by $\mathscr{H} = L^2(\mathscr{I})$ the Hilbert space consisting of all square-integrable surfaces defined on a compact set $\mathscr{I} \subset \mathbb{R}^q$ and equipped with the inner product $\langle f,g\rangle_{\mathscr{H}} = \int_{\mathscr{I}} f(u)g(u)\,du$, for any $f,g \in L^2(\mathscr{I})$. Define the squared L^2 norm of a function by $\|f\|_{\mathscr{H}} = \langle f,f\rangle_{\mathscr{H}}$.

Let $\{Y_i\}_{i=1}^n$ be a series of n random surfaces that take values on $\mathscr{H}_Y = L^2(\mathscr{I}_Y)$. Associated with each Y_i, there is a regressor surface $X_i \in \mathscr{H}_X = L^2(\mathscr{I}_X)$. We consider functions with finite second moment, i.e., $\mathbb{E}[||Y_i||^2_{\mathscr{H}_Y}] < \infty$ and $\mathbb{E}[\|X_i\|^2_{\mathscr{H}_X}] < \infty$. For simplicity, we assume that Y_i and X_i are centered functions, i.e., $\mu_X(v) = \mathbb{E}[X_i(v)] = 0, \forall v \in \mathscr{I}_X$ and $\mu_Y(u) = \mathbb{E}[Y_i(u)] = 0, \forall u \in \mathscr{I}_Y$. Let P_X and P_Y denote the distributions of X and Y, and $P_{Y|X} : \mathscr{H}_X \times \mathscr{H}_Y \to \mathbb{R}$ the conditional distribution of Y given X. If $L_2(P_X)$ represents the class of all measurable functions of X with $\mathbb{E}[f^2(X)] < \infty$ under P_X, then $L_2(P_Y)$ is similarly defined for Y. Our goal is to capture the potential nonlinear dependence between Y_i and X_i through a function $g : \mathscr{H}_X \to \mathscr{H}_Y$

$$Y_i = g(X_i) + \varepsilon_i, \tag{1}$$

where ε_i is a noise function with $\mathbb{E}[\varepsilon_i(u)] = 0, \forall u \in \mathscr{I}_Y$ and $\mathbb{E}[\|\varepsilon_i\|^2_{\mathscr{H}_Y}] < \infty$. In our study, X_i is a vector of lagged surfaces $Y_{i-1}, Y_{i-2}, \ldots$ or their linear combination. Hence, the model (1) is a nonlinear functional autoregression model (NFAR).

We project Y_i onto a set of orthonormal basis functions $\varphi = (\varphi_1, \varphi_2 \ldots)^T$ with $\varphi_j \in \mathscr{H}_Y$

$$Y_i = \sum_{j=1}^{\infty} y_{ij}\varphi_j, \quad \text{with } y_{ij} = \langle Y_i, \varphi_j \rangle_{\mathscr{H}_Y}, \tag{2}$$

with $\boldsymbol{y}_i = (y_{i1}, y_{i2}, ...)^T \in \mathscr{H}_{\boldsymbol{y}} \subseteq \mathbb{R}^\infty$ the projection coefficients of Y_i onto the basis functions φ, satisfying $\mathbb{E}[y_{ij}y_{rv}] = 0$ for $j \neq v$, $j, v \in \mathbb{N}_+$ and any $i, r \in \{1, \ldots, n\}$. Similarly, we project X_i onto a sequence of orthogonal basis functions $\psi = (\psi_{1,}, \psi_2, ...)^T$ with $\psi_j \in \mathscr{H}_X$

$$X_i = \sum_{j=1}^{\infty} x_{ij}\psi_j, \quad \text{with } x_{ij} = \langle X_i, \psi_j \rangle_{\mathscr{H}_X}, \tag{3}$$

with $\boldsymbol{x}_i = (x_{i1}, x_{i2}, ...)^T \in \mathscr{H}_{\boldsymbol{x}} \subseteq \mathbb{R}^\infty$ the projection coefficients of X_i onto the basis functions ψ, satisfying $\mathbb{E}[x_{ij}x_{rv}] = 0$ for $j \neq v$, $j, v \in \mathbb{N}_+$ and any $i, r \in \{1, \ldots, n\}$. Transitioning from functions to vectors, we define $f : \mathscr{H}_x \to \mathscr{H}_y$

$$\boldsymbol{y}_i = f(\boldsymbol{x}_i) + \boldsymbol{\epsilon}_i, \tag{4}$$

where ϵ_i is a noise vector with $\mathbb{E}[\epsilon_{ij}] = 0$ and $\mathbb{E}[||\epsilon_i||^2] < \infty$. In what follows, we estimate f using a multi-task neural network that can be reformulated as a kernel regression in Section 3.2. Next, we demonstrate the connection between the regression in function and vector spaces through a kernel regression in Section 3.3. Although vectors offer a more compact representation of functions, they still exist within an infinite-dimensional framework unless additional restrictions are assumed to hold. This inherent complexity makes the empirical estimation of Equation (4) challenging when working with finite sample sizes. To address this issue, we employ classical sieve methods leading to finite-dimensional vector spaces, as explained in detail in Appendix A.2.[7]

3.2 Neural Tangent Kernel

To estimate Equation (4), we utilize the Neural Tangent Kernel (NTK), a flexible kernel class that uses neural networks to capture complex nonlinear dependencies in data effectively. The NTK describes how neural networks behave under first-order gradient descent training and is calculated as the inner product of the network's weight gradients.

In our study, we apply the NTK to shallow, fully connected networks with wide widths, which have been shown to perform well in finite samples, particularly with weight decay, Lee et al. (2020). Suppose we have a NN of depth L, $f(.;\boldsymbol{\theta}) : \mathbb{R}^{n_0} \to \mathbb{R}^{n_L}$ with parameters $\boldsymbol{\theta}$, where layers are indexed from 0 (input $\boldsymbol{x}$) to L (output $\boldsymbol{y}$), each layer containing $n_0, n_1, ..., n_L$ neurons. We use the NTK parameterization of Jacot, Gabriel, and Hongler (2018) for the NN

$$\begin{aligned}
\text{input layer}: \alpha^{(0)}(\boldsymbol{x};\boldsymbol{\theta}) &= \boldsymbol{x} \in \mathbb{R}^{n_0},\\
\text{preactivation}: \tilde{\alpha}^{(\ell+1)}(\boldsymbol{x};\boldsymbol{\theta}) &= \frac{1}{\sqrt{n_\ell}} W^{(\ell)} \alpha^{(\ell)}(\boldsymbol{x};\boldsymbol{\theta}) + \eta b^{(\ell)},\\
\text{activation}: \alpha^{(\ell+1)}(\boldsymbol{x};\boldsymbol{\theta}) &= \sigma(\tilde{\alpha}^{(\ell+1)}(\boldsymbol{x};\boldsymbol{\theta})),\\
\text{output layer}: f(\boldsymbol{x};\boldsymbol{\theta}) &= \tilde{\alpha}^{(L)}(\boldsymbol{x};\boldsymbol{\theta}) \in \mathbb{R}^{n_L},
\end{aligned}$$

[7] Sieve methods involve truncating the regression for the full set of projection coefficients while striving to minimize any loss of information.

where parameters $\boldsymbol{\theta}$ consist of the connection matrices $W^{(\ell)} \in \mathbb{R}^{n_{\ell+1} \times n_\ell}$, and bias vectors $b^{(\ell)} \in \mathbb{R}^{n_{\ell+1}}$. All parameters are initialized as i.i.d. Gaussians $\mathcal{N}(0,1)$; the constant $\eta > 0$ controls the influence of the bias on the training; and the nonlinear ReLU activation function $\sigma(.)$ is applied element-wise to each element of $\tilde{\alpha}^{(\ell+1)}(\boldsymbol{x};\boldsymbol{\theta})$. Note that NTK models are parameterized differently from the standard NNs, commonly used in previous popular studies in finance, such as Gu, Kelly, and Xiu (2020) and Almeida et al. (2022), which make them suitable for kernel regression.[8] Given a training dataset $\{(\boldsymbol{x}_i, \boldsymbol{y}_i) : i = 1, \ldots, n\}$, the parameters of the NN model are updated and learned by minimizing a least-squares empirical loss function $\mathscr{L}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \|\boldsymbol{y}_i - f(\boldsymbol{x}_i;\boldsymbol{\theta})\|^2$ via the back-propagation algorithm and the Gradient Descent (GD) method with learning rate ζ.

By Theorem 1 and Theorem 2 in Jacot, Gabriel, and Hongler (2018), as $n_1, \ldots, n_L \to \infty$, the NTK parameterized NN converges to the same estimator yielded by kernel regression with the infinite NTK as a kernel. Formally, let the empirical NTK $\tilde{k}^{(L)}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \nabla_{\boldsymbol{\theta}}^T f(\boldsymbol{x}_i;\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} f(\boldsymbol{x}_j;\boldsymbol{\theta})$ be a random matrix computed as the product of the gradients of the NN with respect to the unknown parameters $\boldsymbol{\theta}$ during its training with GD algorithm. Then $k_\infty^{(L)}$ is the limiting scalar kernel that solely depends on the NN architecture, such that for all $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{R}^{n_0}$, $\tilde{k}^{(L)}(\boldsymbol{x}_i, \boldsymbol{x}_j) \xrightarrow{p} k_\infty^{(L)}(\boldsymbol{x}_i, \boldsymbol{x}_j) \otimes I_{n_L}$ as $n_1, n_2, \ldots, n_{L-1} \to \infty$. More details on the NTK can be found in Appendix A.4.

3.3 Kernel regression

To elucidate the nonlinear relation between X_i and Y_i in Equation (1) in connection to the estimation procedure proposed in Section 3.2, we introduce another Hilbert space of functions generated by a positive-definite kernel $K : \mathscr{H}_X \times \mathscr{H}_X \to \mathbb{R}$ defined on the inner product of $\mathscr{H}_X$ through a function $\rho : \mathbb{R}^3 \to \mathbb{R}^+$, such that

$$K(X_i, X_j) = \rho(\langle X_i, X_i \rangle_{\mathscr{H}_X}, \langle X_i, X_j \rangle_{\mathscr{H}_X}, \langle X_j, X_j \rangle_{\mathscr{H}_X}), \tag{5}$$

[8] Usually, the standard NNs do not have the factors $\frac{1}{\sqrt{n_\ell}}$, and their parameters are initialized using LeCun initialization, with $W_{ij}^{\ell} \sim \mathcal{N}(0, 1/n_\ell)$ and $b_j^l \sim \mathcal{N}(0,1)$. The factors $\frac{1}{\sqrt{n_\ell}}$ are essential for obtaining a consistent asymptotic behavior of the NNs as the number of neurons $n_1, n_2, \ldots, n_{L-1}$ go to infinity; while the factor η is introduced to balance the influence of the bias and the weights.

for any $X_i, X_j \in \mathscr{H}_X$. The kernel K satisfies the kernel property $K(X_i, X_j) = \langle K(.,X_i), K(.,X_j)\rangle_{\mathfrak{M}_X}$.[9]

The space induced by K, denoted by $\mathfrak{M}_X$, is a nested space of $\mathscr{H}_X$ via ρ, see Li and Song (2017). For ease of notation, we assume that kernels are demeaned, i.e., for any $X_i \in \mathscr{H}_X$, $\mu_X(X_i) = \langle \mu_X, K(.,X_i)\rangle_{\mathfrak{M}_X} = \mathbb{E}[K(X,X_i)] = 0$. The space $\mathfrak{M}_X$ is called Reproducing Kernel Hilbert Space (RKHS) since $\mathfrak{M}_X = \text{span}\{K(\cdot,X_i) : X_i \in \mathscr{H}_X\}$ and K has the reproducing property, i.e., for any function $g \in \mathfrak{M}_X$, $g(X_i) = \langle g, K(\cdot,X_i)\rangle_{\mathfrak{M}_X}$.

The introduction of the kernel K is crucial in capturing the essence of the underlying nonlinear relationship. With the RKHS $\mathfrak{M}_X$ generated by K, the nonlinear function in the model is represented by a linear expansion of functions in the nested space. Let $\mathscr{B}(\mathscr{H}_1, \mathscr{H}_2)$ be the class of bounded linear operators mapping a Hilbert space $\mathscr{H}_1$ to another Hilbert space $\mathscr{H}_2$. Then $Bg(X_i) = \langle Bg, K(\cdot,X_i)\rangle_{\mathfrak{M}_X} = \langle g, B^*K(\cdot,X_i)\rangle_{\mathscr{H}_Y}$, for $g \in \mathscr{H}_Y, B \in \mathscr{B}(\mathscr{H}_Y, \mathfrak{M}_X)$ and $B^* \in \mathscr{B}(\mathfrak{M}_X, \mathscr{H}_Y)$ the adjoint operator of B. This means that we can represent functions in $\mathscr{H}_Y$ by means of the kernel.

The function-on-function regression problem in Equation (1) can be reformulated as a functional kernel regression, in which the task is to find $B_0 \in \mathscr{B}(\mathscr{H}_Y, \mathfrak{M}_X)$ such that

$$B_0 = \underset{B \in \mathscr{B}(\mathscr{H}_Y, \mathfrak{M}_X)}{\arg\min} \; \mathbb{E}[\|Y_i - B^*K(.,X_i)\|^2_{\mathscr{H}_Y}]. \tag{6}$$

This model can be viewed as an extension of the traditional multivariate linear model that associates vector responses with vector covariates. The functional normal equation of the least squares regression from the RKHS to $\mathscr{H}_Y$ takes the form $\Sigma_{XY} = \Sigma_{XX}B_0$, with the (cross-)covariance operators $\Sigma_{XX} \in \mathscr{B}(\mathfrak{M}_X, \mathfrak{M}_X)$, $\Sigma_{XY} \in \mathscr{B}(\mathscr{H}_Y, \mathfrak{M}_X)$ such that $\Sigma_{XX} = \mathbb{E}[K(.,X) \otimes K(.,X)]$ and $\Sigma_{XY} = \mathbb{E}[K(.,X) \otimes Y]$. Since Σ_{XX} is a compact operator in L^2, its inverse is not bounded, leading to an ill-posed problem. To address this issue, we define $\Sigma^\dagger_{XX}$ to be the Moore-Penrose inverse of Σ_{XX}. Theorem 2.1. in Sang and Li (2022) states that a solution to the regression (6) is given by

$$B_0 = \Sigma^\dagger_{XX}\Sigma_{XY}. \tag{7}$$

[9] Intuitively, $K(.,X_i) : \mathscr{H}_X \to \mathfrak{M}_X$ can be thought of as a feature map defined by the kernel, and $K(X_i,X_j)$ as a measure of similarity between any two curves X_i, X_j in the $\mathfrak{M}_X$ space.

The solution B_0 is well-defined under the following additional assumptions:

Assumption 1. *$\mathfrak{M}_X$ is a dense subset of $L_2(P_X)$ and $\mathbb{E}[|Y\|^2_{\mathscr{H}_Y}] < \infty$;*

Assumption 2. *There exists a constant $C > 0$ so that for any $f \in \mathfrak{M}_X$, $\mathbb{E}[f^2(X)] \leq C\|f\|^2_{\mathfrak{M}_X}$;*

Assumption 3. *$ran(\Sigma_{XY}) \subseteq ran(\Sigma_{XX})$ and $\Sigma^{\dagger}_{XX}\Sigma_{XY}$ is a bounded operator.*

Assumptions 1 – 3 are typical in the functional kernel regression literature and have been utilized in previous studies, such as by Li and Song (2017), Fukumizu, Bach, and Jordan (2009), and Sang and Li (2022). Assumption 1 ensures the existence of a mapping between any target measurable function and a sequence of functions in the nested space, while Assumption 2 guarantees that this mapping is bounded, ensuring stability. Assumption 3 restricts the linear operator that maps the variable Y to $K(.,X)$, requiring it to be bounded. Under these assumptions, Proposition 2.2. and Proposition 2.3. of Sang and Li (2022) lead to the following relation between B_0 and the predicted value of Y_i for a given $X_i \in \mathscr{H}_X$ and the kernel K

$$\begin{aligned}\mathbb{E}[Y_i|X_i] &= B_0^{*}K(.,X_i)\\ &= \Sigma_{YX}\Sigma^{\dagger}_{XX}K(.,X_i)\\ &= \mathbb{E}[\{(\Sigma^{\dagger}_{XX}K(.,X_i))(X)\}Y].\end{aligned} \tag{8}$$

where $B_0^{*} = \Sigma_{YX}\Sigma^{\dagger}_{XX} \in \mathscr{B}(\mathfrak{M}_X, \mathscr{H}_Y)$ is the adjoint operator of B_0, and $\Sigma_{YX} = \Sigma^{*}_{XY} = \mathbb{E}[Y \otimes K(.,X)] \in \mathscr{B}(\mathfrak{M}_X, \mathscr{H}_Y)$. The last equality is an expectation of weighted function Y, where the random weights are defined as $W : \mathscr{H}_X \times \mathscr{H}_X \to \mathbb{R}$, $W(X_i, X) := (\Sigma^{\dagger}_{XX}K(.,X_i))(X) = \langle \Sigma^{\dagger}_{XX}K(.,X_i), K(.,X)\rangle_{\mathfrak{M}_X}$.

3.3.1 From RKHS of functions to RKHS of vectors

We define a new kernel $k : \mathscr{H}_{\boldsymbol{x}} \times \mathscr{H}_{\boldsymbol{x}} \to \mathbb{R}$ such that for any $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathscr{H}_{\boldsymbol{x}}$

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \rho(\langle \boldsymbol{x}_i, \boldsymbol{x}_j\rangle, \langle \boldsymbol{x}_i, \boldsymbol{x}_j\rangle, \langle \boldsymbol{x}_j, \boldsymbol{x}_j\rangle). \tag{9}$$

Denote $\mathfrak{M}_{\boldsymbol{x}}$ the RKHS induced by k and determined by the inner product of $\mathscr{H}_{\boldsymbol{x}}$. The following theorem states the isomorphism between the two Reproducing Kernel Hilbert Spaces $\mathfrak{M}_X$ and $\mathfrak{M}_{\boldsymbol{x}}$.

Lemma 1 (**Isomorphism between Reproducing Kernel Hilbert Spaces**). *Under Equations* (3) *and* (9)*, it holds that*

$$\begin{aligned} k(\boldsymbol{x}_i, \boldsymbol{x}_j) &= \langle k(.,\boldsymbol{x}_i), k(.,\boldsymbol{x}_j) \rangle \\ &= \langle K(.,X_i), K(.,X_j) \rangle_{\mathfrak{M}_X} = K(X_i, X_j). \end{aligned} \tag{10}$$

Then the RKHS $\mathfrak{M}_X$ nested on $\mathscr{H}_X$ is isometrically isomorphic to the RKHS $\mathfrak{M}_{\boldsymbol{x}}$ nested on $\mathscr{H}_{\boldsymbol{x}}$.

Lemma 1 is closely related to Lemma 4.2 from Klepsch and Klüppelberg (2017), which demonstrates the isometric isomorphism between a functional space and its corresponding vector space when projecting functional objects onto orthogonal bases. We extend this result to functions in RKHS, showing that $\mathfrak{M}_X$ is isometrically isomorphic to $\mathfrak{M}_{\boldsymbol{x}}$.

3.3.2 From function-to-function to vector-to-vector regression

Let $\Sigma_{\boldsymbol{x}\boldsymbol{x}} = \mathbb{E}\big[k(.,\boldsymbol{x}) \otimes k(.,\boldsymbol{x})\big]$ be the covariance matrix of $k(.,\boldsymbol{x})$ and $\Sigma^{\dagger}_{\boldsymbol{x}\boldsymbol{x}}$ its Moore-Penrose inverse. Further define $\boldsymbol{y} = (y_{i1}, y_{i2}, \ldots)^T \in \mathscr{H}_{\boldsymbol{y}} \subseteq \mathbb{R}^{\infty}$, and $\Sigma_{\boldsymbol{y}\boldsymbol{x}} = \mathbb{E}[\boldsymbol{y} \otimes k(.,\boldsymbol{x})]$. Now, we can establish the link between function-to-function regression and vector-to-vector regression.

Theorem 1 (**Vector-to-vector regression**). *Given the decomposition of Y_i in Equation* (2) *and X_i in Equations* (3)*, under Assumptions* (1) - (3) *and Lemma 1, for a positive definite kernel k defined by Equation* (9)*, if there is a covariance matrix $\Sigma_{\boldsymbol{x}\boldsymbol{x}}$ of $k(.,\boldsymbol{x})$ that is diagonal, then the function-to-function regression model in Equation* (6) *may be represented equivalently by*

$$\beta_0 = \underset{\beta \in \mathscr{B}(\mathscr{H}_{\boldsymbol{y}}, \mathfrak{M}_{\boldsymbol{x}})}{\arg\min} \mathbb{E}[\|\boldsymbol{y}_i - \beta^* k(.,\boldsymbol{x}_i)\|^2], \tag{11}$$

with solution $\beta_0 = \Sigma_{\boldsymbol{xx}}^{\dagger}\Sigma_{\boldsymbol{xy}}$. This leads to

$$\begin{aligned}
\mathbb{E}[\boldsymbol{y}_i|\boldsymbol{x}_i] &= \beta_0^* k(.,\boldsymbol{x}_i) \\
&= \Sigma_{\boldsymbol{yx}}\Sigma_{\boldsymbol{xx}}^{\dagger} k(.,\boldsymbol{x}_i) \qquad (12)\\
&= \mathbb{E}[\{(\Sigma_{\boldsymbol{xx}}^{\dagger} k(.,\boldsymbol{x}_i))(\boldsymbol{x})\}\boldsymbol{y}].
\end{aligned}$$

The theorem presents the equivalence between functional kernel regression and vector kernel regression under Assumptions (1) - (3), and Lemma 1. This result is a novel contribution to the literature. While Cho et al. (2013) established the equivalence between functional linear regression and vector linear regression, to the best of our knowledge, we are the first to demonstrate this equivalence for nonlinear kernel regression in both functional and vector spaces. It also shows that a functional kernel regression leads to $\mathbb{E}[Y_i|X_i] = \varphi^T \mathbb{E}[\boldsymbol{y}_i|\boldsymbol{x}_i]$ under suitable conditions. Since NTK can be reformulated as a kernel regression in vector spaces, the conditional estimators correspond to an NTK regression.

3.4 fNTK Algorithm

We develop an fNTK algorithm for the NFAR model, which is illustrated in Figure 3 and detailed in Algorithm 1. First, we project functional observations Y_i and X_i onto some orthonormal basis functions as described in Section 3.1. In the next step, we employ the projection coefficients in a multi-task NN parametrized as an NTK described in 3.2. Finally, the NTK estimator is used to evaluate new target functions. In practice, empirical coefficients are retrieved from a finite dataset with n observations. For computational feasibility, we consider finite-dimensional vector spaces by employing the sieve method. We also conduct a simulation study in Appendix A.5, showing that the proposed estimation is numerically consistent.

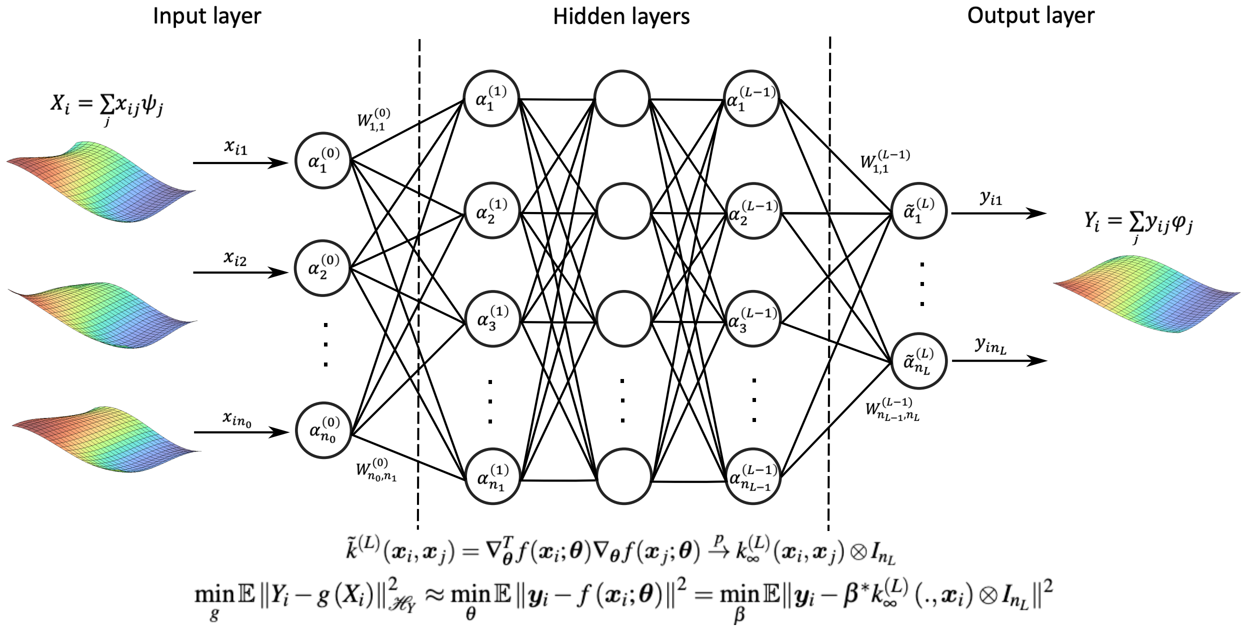

Input layer
Hidden layers
Output layer
$X_i = \sum_j x_{ij}\psi_j$
x_{i1}
x_{i2}
x_{in_0}
$W_{1,1}^{(0)}$
$W_{n_0,n_1}^{(0)}$
$\alpha_1^{(0)}$
$\alpha_2^{(0)}$
$\alpha_{n_0}^{(0)}$
$\alpha_1^{(1)}$
$\alpha_2^{(1)}$
$\alpha_3^{(1)}$
$\alpha_{n_1}^{(1)}$
$\alpha_1^{(L-1)}$
$\alpha_2^{(L-1)}$
$\alpha_3^{(L-1)}$
$\alpha_{n_{L-1}}^{(L-1)}$
$W_{1,1}^{(L-1)}$
$W_{n_{L-1},n_L}^{(L-1)}$
$\tilde{\alpha}_1^{(L)}$
$\tilde{\alpha}_{n_L}^{(L)}$
y_{i1}
y_{in_L}
$Y_i = \sum_j y_{ij}\varphi_j$
$\tilde{k}^{(L)}(\boldsymbol{x}_i,\boldsymbol{x}_j) = \nabla_{\boldsymbol{\theta}}^T f(\boldsymbol{x}_i;\boldsymbol{\theta})\nabla_{\boldsymbol{\theta}} f(\boldsymbol{x}_j;\boldsymbol{\theta}) \xrightarrow{P} k_\infty^{(L)}(\boldsymbol{x}_i,\boldsymbol{x}_j)\otimes I_{n_L}$
$\min_g \mathbb{E}\,\|Y_i - g(X_i)\|^2_{\mathscr{H}_Y} \approx \min_\theta \mathbb{E}\,\|\boldsymbol{y}_i - f(\boldsymbol{x}_i;\boldsymbol{\theta})\|^2 = \min_\beta \mathbb{E}\|\boldsymbol{y}_i - \boldsymbol{\beta}^* k_\infty^{(L)}(.,\boldsymbol{x}_i)\otimes I_{n_L}\|^2$

Figure 3: Training an fNTK parametrized network $f(\boldsymbol{x}_i;\boldsymbol{\theta})$. Parameters $\boldsymbol{\theta} = \left\{W^{(\ell)}, b^{(\ell)} : \ell = 1, L-1\right\}$ are initialized as i.i.d. $\mathscr{N}(0,1)$ and updated in each step of the gradient descent algorithm. The NTK network converges to the same estimator yielded by kernel regression with the infinite NTK as a kernel.

Algorithm 1 fNTK regression for IVS forecasting

```
1. For $i \in \{1,\ldots,n\}$, the response $Y_i = IV_{i+h}$ is the IVS of day $i+h$, $h \in \mathbb{N}_+$, and the predictor $X_i = (IV_i^{(d)}, IV_i^{(w)}, IV_i^{(m)})^T$ contains daily ($d$), weekly ($w$) and monthly ($m$) IVS lags.

2. Project $Y_i$ and $X_i$ onto a finite sequence of $J$ orthonormal spline basis functions $\{\varphi_j\} = \{\psi_j\}$. Denote $\boldsymbol{y}_i = (y_{i,1},\ldots,y_{i,J})^T$ the vector of basis coefficients of $Y_i$, and $\boldsymbol{x}_i = (\boldsymbol{x}_i^{(d)}, \boldsymbol{x}_i^{(w)}, \boldsymbol{x}_i^{(m)})^T$, the matrix of basis coefficients of $X_i$, with $\boldsymbol{x}_i^{(\ell)} = (x_{i,1}^{(\ell)},\ldots,x_{i,J}^{(\ell)})^T$, $\ell \in \{d,w,m\}$.

3. Train an NTK parameterized neural network $f(\boldsymbol{x};\boldsymbol{\theta})$ on $\{(\boldsymbol{y}_i,\boldsymbol{x}_i)\}_{i=1}^n$ to find $\hat{\theta}$ that minimizes the empirical least-square loss function $\mathscr{L}(\boldsymbol{\theta}) = \sum_{i=1}^n \|\boldsymbol{y}_i - f(\boldsymbol{x}_i;\boldsymbol{\theta})\|^2$.

4. For out-of-sample prediction, take a new functional observation $X'$, project it onto the spline basis to obtain coefficients $\boldsymbol{x}'$, then predict $\hat{\boldsymbol{y}}' = f(\boldsymbol{x}';\hat{\theta})$ and $\hat{Y}' = \sum_{j=1}^J \hat{y}'_j\varphi_j$.
```

4 Forecasting Implied Volatility Surfaces

This section aims to investigate the effectiveness of the proposed neural-based machine learning estimation approach, fNTK, for nonlinear functional autoregressive models in forecasting the IVS. We compare the fNTK approach with classical IV forecasting alternatives and nonparametric functional estimation using alternative kernels to assess their predictive performance.

4.1 Forecasting Framework

Let IV_i be the IVS of day i, and $\varphi = \{\varphi_j : j = 1,2,...,J\}$ be a set of two-dimensional orthogonal splines functions of Redd (2012). Specifically, we utilize the orthogonal cubic splines with intercepts and no interior knots, which leads to the total number of basis functions $J = 16$.[10] In a h-step ahead forecasting, the response vector $\boldsymbol{y}_i$ represents the basis coefficients of the future implied volatility surface $Y_i = IV_{i+h}$ of day $i+h$. To forecast $\boldsymbol{y}_i$, we incorporate basis coefficients of different lags of implied volatility surfaces using a restricted functional VAR framework. Let $\boldsymbol{x}_i^{(d)}$, $\boldsymbol{x}_i^{(w)}$, and $\boldsymbol{x}_i^{(m)}$ be the basis coefficients of $X_i^{(d)} = IV_i$, $X_i^{(w)} = \frac{1}{5}\sum_{k=i-4}^{i} IV_k$ and $X_i^{(m)} = \frac{1}{22}\sum_{k=i-21}^{i} IV_k$, then we have $\boldsymbol{x}_i = (\boldsymbol{x}_i^{(d)}, \boldsymbol{x}_i^{(w)}, \boldsymbol{x}_i^{(m)})^T$. This approach effectively captures the daily, weekly, and monthly features of the IVS, or the three primary volatility components associated with different types of traders: short-term, medium-term, and long-term, see Zhang, Li, and Zhang (2023).[11] Both $\boldsymbol{y}_i$ and $\boldsymbol{x}_i$ are standardized before training the models. It is important to note that standardization is based solely on the training data to prevent information leakage from the test data.

The Nonlinear Functional Autoregressive (NFAR) model is trained using a rolling window spanning 2500 days, ensuring that the model remains adaptable to the most recent market conditions. This training practice results in a roughly 80%-20% train-test split ratio. We update hyperparameters every six months and adjust parameters daily for out-of-sample forecasting. Initially, the model is trained on data from the past 2500 days, divided into a 2000-day training set and a 500-day validation set. We select the hyperparameters that minimize the RMSE on the validation set, and these values are then used as hyperparameters. These hyperparameters are applied daily to update the model using a moving window of training data over the following six months. At the end of this period, we reassess and adjust the hyperparameters as necessary. Detailed values of

[10] We experimented with different numbers of equidistant knots, such as a single knot at $1/2$ and two knots at $\{1/3, 2/3\}$. The RMSE for the smoothed surfaces remained largely consistent across these variations. In addition, we investigated the robustness of our framework for various degrees of splines: 2, 4, 5, and 6, which correspond to the total number of basis functions $J = 9, 25, 36,$ and 49 respectively, see Appendix A.9.1.

[11] The effect of using a restricted VAR versus using all 22 lags unrestrictedly is reported in Appendix A.9.1.

hyperparameters are provided in Appendix A.7.

Additionally, we perform a robustness check by training models using rolling windows of 1000 and 2000 days, further partitioned into 500/500 and 1500/500 days for training and validation, respectively. The results closely align with the reported findings and are accessible for reference in Appendix A.9.

4.2 Alternative Models

In addition to fNTK, we explore alternative nonparametric functional estimation approaches. We consider a kernel regression approach, using linear, Gaussian, and Laplacian kernels within the proposed NFAR framework. The linear kernel (LinK) is defined by $k(\boldsymbol{x},\boldsymbol{x}') = \boldsymbol{x}^T\boldsymbol{x}'$. In this case, the new RKHS is the same as the original Hilbert input variable space, and the kernel regression is equivalent to a linear regression. In other words, it becomes functional autoregression. We also utilize two popular nonlinear parametric kernels: the Gaussian kernel (GauK), also known as the radial basis function kernel ($k(\boldsymbol{x},\boldsymbol{x}') = \exp(-\gamma\|\boldsymbol{x}-\boldsymbol{x}'\|^2)$) and the Laplacian kernel (LapK) ($k(\boldsymbol{x},\boldsymbol{x}') = \exp(-\gamma\|\boldsymbol{x}-\boldsymbol{x}'\|_1)$), where γ is a constant controlling the roughness of the kernel.[12] For the parametric kernels, exact solutions are obtained in a kernel-ridge regression as explained in Appendix A.6.

We also consider several classical models for implied volatility forecasting. The objective is to scrutinize the importance of nonparametric modeling in capturing nonlinear dynamics. The classical models encompass the renowned Carr and Wu model by Carr and Wu (2016) and the Ad-Hoc Black-Scholes model by Dumas, Fleming, and Whaley (1998). These models have demonstrated effectiveness for S&P 500 IVS and were previously employed in studies by Goncalves and Guidolin (2006), Bernales and Guidolin (2014), and Almeida et al. (2022). Additionally, we adapt the random walk model to our functional framework. More details on the alternative models can be found in Appendix A.8.

[12] We experimented with polynomial kernels of different degrees of 2, 3, 4, and 5. However, their performance is significantly worse than all other kernels; hence, our methodology and results do not report it.

Random Walk with Deep Neural Network (DNN-RW) Model. A neural network (NN) is trained on implied volatilities of observed options on a particular day (i), with time-to-maturity and moneyness serving as inputs and implied volatility as output. The trained NN is then utilized to predict the implied volatilities of a future day ($i+h$). This model is equivalent to first fitting the Black-Scholes model, which predicts a flat surface, and then using an NN to adjust the main curvatures of the actual implied volatility surface.

Carr and Wu (CW) Model. The approach introduced by Carr and Wu (2016) offers an option pricing framework that characterizes implied volatility dynamics across various strikes and maturities. For an option with strike and time to maturity, the risk-neutral measures encapsulate the dynamics of the underlying spot price and the option implied volatility. The model involves the instantaneous variance rate, the average implied volatility drift, and the exponential dampening parameters. We employ the relative strike and formulate a quadratic equation for the square implied volatility. Parameter estimation involves minimizing a nonlinear least squares problem, and predictions are derived by solving the quadratic Equation using the estimated parameters and either a random walk (CW-RW) or a correction with a deep neural network (CW-DNN) model.

Ad-Hoc Black–Scholes (AHBS) Model. Features of a cross-section of options on moneyness and maturities are extracted based on the ad-hoc Black-Scholes (AHBS) model by Dumas, Fleming, and Whaley (1998). The features capture different characteristics of the surface: the moneyness (smile/skew) slope, the curvature in the moneyness dimension, the maturity (term-structure) slope, the curvature in the maturity dimension, and the interactions between the moneyness and time-to-maturity dimensions. Three models, including a random walk (AHBS-RW) model, a correction with a deep neural network (AHBS-DNN) model, and a vector autoregressive (AHBS-VAR) model, are then utilized to predict the AHBS features, enabling the forecasting of implied volatilities.

Autoencoder with Long Short-Term Memory (AE-LSTM) Model. We adapted the modeling framework proposed by Zhang, Li, and Zhang (2023), which utilizes an autoencoder (AE) to extract latent features from the interpolated implied volatility (IV) grid values for each day, and a long short-term memory (LSTM) network to capture temporal dependencies among these features.[13] To ensure a fair comparison with our proposed framework, we use cubic orthogonal splines for IV interpolation, replacing the AHBS used in the original work.

Functional Random Walk (fRW) Model. The functional random walk model, adapted from Bernales and Guidolin (2014), predicts the implied volatility surface of a future day to be the same as the current day's. This straightforward approach is a comparative baseline to gauge the significance of incorporating implied volatility surface dynamics in enhancing predictions.

4.3 Forecasting Performance Measures

We assess prediction accuracy and goodness of fit derived from observed test data. Specifically, root mean square error (RMSE) captures prediction accuracy, and out-of-sample R^2 (OoR^2) measures the goodness of fit, depicting the proportion of variance the models explain[14]

$$\mathrm{RMSE}_h = \sqrt{\frac{1}{\sum_{i=i_0}^{n-h} n_i} \sum_{i=i_0}^{n-h} \sum_{j=1}^{n_i} (Y_i(\tau_j, m_j) - \hat{Y}_i(\tau_j, m_j))^2},$$

$$\mathrm{Oo}R_h^2 = 1 - \frac{\sum_{i=i_0}^{n-h} \sum_{j=1}^{n_i} (Y_i(\tau_j, m_j) - \hat{Y}_i(\tau_j, m_j))^2}{\sum_{i=i_0}^{n-h} \sum_{j=1}^{n_i} (Y_i(\tau_j, m_j) - \bar{Y}_i)^2}.$$

Here, n_i denotes the number of observed options on day i, and $i_0 = 2522$ and $n = 3273$ mark the start and end of the testing period, respectively.

4.4 Results

Table 1 and Figure 4 provide an overview of model performance across different forecasting horizons ($h = 1, 5, 10, 15$, and 20 days ahead).[15] For one-step-ahead predictions, we note that classical models, such as AHBS-VAR and fLink, outperform fNTK in certain instances. Specifically,

[13] The comparison between latent feature extraction using AE and principal component analysis (PCA) can be found in Appendix A.9.4.

[14] In addition, we report mean absolute percentage error (MAPE) and mean correct prediction of the direction of change (MCPDC) in the Appendix A.9.

[15] For clarity, it is worth noting that the results reported represent the average performance measures for both put and call options. Detailed results for put and call options are available upon request.

AHBS-VAR, CW-DNN, and fLink statistically significantly perform better than fNTK at the 5% significance level in the one-sided Diebold-Mariano test based on mean squared errors, as highlighted in Table 1.[16] This can be attributed to the highly persistent patterns and linear relationships in the lag-1 temporal dependence, which are better captured by linear models or random walk predictors. However, as the forecasting horizon extends, the dynamics change notably. The fNTK model demonstrates substantial performance gains for extended prediction horizons, surpassing all other models. For instance, in the 20-day-ahead forecast, fNTK outperforms the classical fRW and AHBS-VAR models by approximately 40%.[17] Moreover, none of the alternative models are statistically significantly better than fNTK at the 5% significance level in the Diebold-Mariano tests for forecasting horizons longer than one day. In addition to RMSE and OoR^2, we provide insights into the model performance through mean absolute percentage error (MAPE) and mean correct prediction of the direction of change (MCPDC) in Table A.3. These metrics exhibit similar patterns to RMSE, with fNTK consistently outperforming parametric and classical models. These findings remain robust when varying the training sample size to 1000 or 2000 days, see Table A.7.

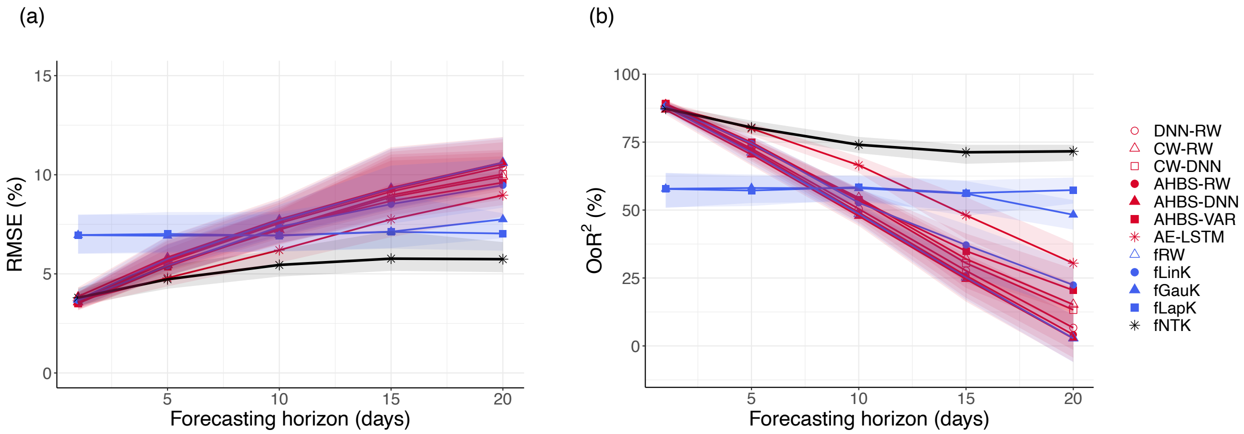


Figure 4: Prediction measurement in terms of (a) RMSE, (b) OoR^2 of all models for forecasting horizons $h = 1, 5, 15, 10$ and 20. The prediction period is from Jan 09, 2019 to Dec 31, 2021. Shaded areas are the 95% confidence intervals using block bootstrapping.

To ensure robustness, we also analyze model performance on an annual basis, as presented in

[16] More details on the Diebold-Mariano test can be found in the Appendix A.9.2.

[17] The 95% confidence intervals are obtained using the non-overlapping block bootstrap method (Härdle, Horowitz, and Kreiss, 2003), with a bootstrap length of 20 days. Varying the bootstrap lengths does not significantly impact the confidence intervals. Results with bootstrap lengths of 10, 40, and 80 are available upon request.

	RMSE					OoR^2				
	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$
Overall (from Jan 9, 2019 to Dec 31, 2021)										
DNN-RW	3.62	5.61	7.56	9.15	10.40	88.45	72.62	50.43	27.68	6.67
CW-RW	*3.57	5.45	7.23	8.87	9.91	88.74	74.18	54.50	32.01	15.21
CW-DNN	****3.51**	5.47	7.29	8.97	10.03	**89.11**	74.00	53.69	30.47	13.27
AHBS-RW	3.63	5.68	7.65	9.26	10.54	88.39	71.90	49.17	25.87	4.16
AHBS-DNN	3.86	5.83	7.75	9.34	10.62	86.91	70.39	47.81	24.60	2.76
AHBS-VAR	****3.51**	5.37	7.30	8.68	9.60	**89.11**	74.93	53.72	34.93	20.49
AE-LSTM	3.65	4.80	6.20	7.75	8.97	88.27	79.98	66.51	48.08	30.44
fRW	3.71	5.75	7.69	9.30	10.61	87.92	71.23	48.57	25.23	2.84
fLinK	**3.66	5.41	7.37	8.52	9.47	88.20	74.51	52.70	37.16	22.42
fGauK	6.95	6.92	6.95	7.13	7.75	57.84	58.11	58.02	56.05	48.27
fLapK	6.96	7.01	6.93	7.12	7.04	57.77	57.19	58.42	56.22	57.35
fNTK	3.80	**4.73**	**5.45**	**5.77**	**5.74**	87.31	**80.39**	**74.02**	**71.26**	**71.62**
From Jan 9, 2019 to Dec 31, 2019										
DNN-RW	1.59	2.36	2.82	3.10	3.40	85.11	67.05	52.92	43.13	32.33
CW-RW	1.53	2.23	2.65	2.91	3.19	86.13	70.52	58.33	50.05	40.40
CW-DNN	**1.48	2.23	2.66	2.92	3.21	87.02	70.66	58.09	49.56	39.52
AHBS-RW	1.59	2.42	2.90	3.19	3.50	85.15	65.51	50.36	39.82	28.10
AHBS-DNN	1.67	2.47	2.94	3.23	3.54	83.56	64.04	49.00	38.45	26.61
AHBS-VAR	1.54	2.25	2.65	2.87	2.95	86.06	70.20	58.44	51.20	48.82
AE-LSTM	1.62	2.30	2.55	2.90	2.89	84.48	68.53	61.54	50.09	50.97
fRW	1.53	2.40	2.91	3.21	3.53	86.20	65.90	50.04	39.24	27.12
fLinK	*****1.41**	2.18	2.48	2.75	2.87	**88.19**	72.02	63.56	55.33	51.82
fGauK	**1.46	2.02	2.26	2.35	2.26	87.36	75.92	69.66	67.29	69.66
fLapK	**1.46	***1.88**	2.00	**2.01**	**1.93**	87.39	**79.03**	76.33	**76.10**	**78.09**
fNTK	1.57	2.02	**1.99**	2.07	2.08	85.51	75.98	**76.63**	74.45	74.32
From Jan 1, 2020 to Dec 31, 2020										
DNN-RW	5.27	8.24	11.36	13.91	15.88	83.05	58.74	19.58	-22.12	-62.13
CW-RW	5.29	8.11	10.95	13.57	15.19	82.92	59.91	24.89	-16.35	-48.49
CW-DNN	5.22	8.15	11.06	13.74	15.38	83.31	59.51	23.37	-19.17	-52.08
AHBS-RW	5.34	8.36	11.51	14.08	16.08	82.61	57.48	17.59	-25.09	-66.39
AHBS-DNN	5.70	8.60	11.66	14.20	16.20	80.27	55.01	15.39	-27.17	-68.72
AHBS-VAR	***5.17**	7.92	11.03	13.22	14.67	**83.64**	61.85	24.33	-10.26	-38.14
AE-LSTM	5.39	6.97	9.23	11.81	13.77	82.19	70.40	46.68	11.83	-22.81
fRW	5.52	8.48	11.58	14.15	16.20	81.42	56.22	16.47	-26.39	-69.00
fLinK	5.50	8.03	11.17	12.98	14.51	81.43	60.71	21.86	-7.09	-36.35
fGauK	11.01	10.77	10.71	10.90	11.92	26.97	28.44	28.29	24.79	8.91
fLapK	11.02	10.97	10.76	11.00	10.86	26.88	26.43	28.29	23.71	24.25
fNTK	5.70	**7.07**	**8.29**	**8.71**	**8.67**	80.17	**69.00**	**56.61**	**52.14**	**51.77**
From Jan 1, 2021 to Dec 31, 2021										
DNN-RW	2.20	3.18	3.61	3.81	3.93	87.14	73.22	65.44	61.29	57.10
CW-RW	1.97	2.81	3.22	3.40	3.49	89.69	78.95	72.39	69.14	66.14
CW-DNN	1.89	2.80	3.21	3.40	3.49	90.52	79.26	72.53	69.18	66.18
AHBS-RW	2.08	3.17	3.64	3.85	3.97	88.51	73.34	64.79	60.55	56.23
AHBS-DNN	2.14	3.21	3.68	3.89	4.01	87.74	72.63	64.05	59.81	55.47
AHBS-VAR	2.00	2.96	3.38	3.58	3.69	89.39	76.70	69.75	65.88	62.21
AE-LSTM	1.99	2.78	3.15	2.98	3.09	89.45	79.47	73.64	76.40	73.47
fRW	1.98	3.15	3.64	3.84	3.96	89.64	73.74	64.74	60.84	56.57
fLinK	***1.85**	2.92	3.43	3.55	3.50	**90.94**	77.45	68.75	66.41	66.06
fGauK	*1.86	2.43	2.56	2.82	2.70	90.85	84.32	82.59	78.91	79.86
fLapK	***1.85**	***2.27**	**2.32**	**2.49**	**2.31**	**90.94**	**86.35**	**85.69**	**83.51**	**85.24**
fNTK	1.90	2.38	2.35	2.61	2.43	90.44	84.95	85.34	81.84	83.63

Table 1: Prediction accuracy for all models in different forecasting horizons. Results are for four prediction periods: overall (from Jan 09, 2019 to Dec 31, 2021), from Jan 09, 2019 to Dec 31, 2019, from Jan 01, 2020 to Dec 31, 2020, and from Jan 01, 2020 to Dec 31, 2020. Bold numbers indicate the best-performing model (or models) in a given column. One, two, and three asterisks denote that the alternative model significantly outperforms fNTK according to the Diebold-Mariano test based on mean squared errors, at the 10%, 5%, and 1% significance levels, respectively.

Table 1. The consistent superiority of nonlinear models for extended forecasting horizons is evident across three distinct periods: 2019, the year prior to the Covid-19 pandemic; 2020, marked by pandemic-induced market volatility; and 2021, a year of market recovery. Remarkably, nonlinear models, with fNTK leading the way, exhibit enhanced performance relative to fLinK, AHBS, CW, and fRW, especially during the turbulent year of 2020. The performance gap widens during periods of heightened market volatility, as exemplified by 2020. For instance, in the 20-day ahead prediction scenario in 2020, fNTK demonstrates an $\mathrm{Oo}R^2$ approximately 170% to 240% higher than that of linear and classical models, underscoring its dominant performance during times of increased uncertainty. We further investigate the robustness of these models with respect to hyperparameters, moneyness, and maturity groups, as detailed in Appendix A.9.1. These findings collectively emphasize the robustness of nonlinear models, particularly fNTK, in capturing the intricate dynamics of implied volatility surfaces across diverse market conditions.

5 Economic Value of Predictability

This section investigates the economic value of an accurate IV forecaster. By leveraging the enhanced predictability offered by the fNTK model, we demonstrate its potential benefits for investors in the options market. Our evaluation encompasses various trading strategies: short call and put delta-hedging and delta-neutral straddle strategies,[18] that exploit the h-step ahead forecasts of IVS. We also conduct robustness checks under different test periods, transaction costs, and varying filtering thresholds in the trading.

5.1 Trading Strategies

The trading strategies we consider enable option traders to focus on fluctuations in volatility. The delta hedging strategy is specifically designed to mitigate the risk associated with underlying asset price movements by establishing a corresponding position in an option and the asset. Similarly, the

[18] The results for long call and put delta-hedging and long delta-neutral straddle strategies are available upon request.

delta-neutral straddle strategy relies on option deltas and attains directional neutrality by balancing opposite exposures in the underlying and simultaneously investing in a weighted combination of call and put options with the same characteristics.[19] Our setup relies on trading signals extracted from the predicted IVS and includes only non-zero volume options at time i. An increase (decrease) in the implied volatility is equivalent to an increase (decrease) in the options prices. For brevity, we describe the delta-neural straddle strategies below and delegate the details of delta-hedging strategies to Appendix A.10.

For the short delta-neutral straddle strategy, on each day i, we short w_i units of call option and $(1-w_i)$ units of put option of the same strike and expiration date. Denote Δ_i^c and Δ_i^p the Black-Scholes delta of call and put option, respectively. The weights $w_i = -\Delta_i^p/(\Delta_i^c - \Delta_i^p)$ and $1-w_i$ are used to ensure the straddle delta equal 0. We trade a pair of put and call options if the IV of both options is predicted to decrease on the day $i+h$, see Gao, Xing, and Zhang (2018). Recall that put and call IVS are modeled and predicted separately. Define Q_i as the set of put and call option pairs to be traded, and let $C_{j,i}$ and $P_{j,i}$ be the price of the call and put option, respectively. On day i, we sell all the pairs of options in Q_i and gain a cash inflow of $\sum_{j\in Q_i}(w_{j,i}C_{j,i} + (1-w_{j,i})P_{j,i})$, and close off the positions by paying a cost of $\sum_{i\in Q_i}(w_{j,i}C_{j,i+h} + (1-w_{j,i})P_{j,i+h})$ on day $i+h$. The returns R_i of the short delta-neutral straddle portfolios is

$$R_i = 1 - \frac{\sum_{j\in Q_i}(w_{j,i}C_{j,i+h} + (1-w_{j,i})P_{j,i+h})}{\sum_{j\in Q_i}(w_{j,i}C_{j,i} + (1-w_{j,i})P_{j,i})}$$

We use a filtering threshold of 0.5% deviation in implied volatility for each trading strategy. Specifically, on a given day i, if the implied volatility of an option is predicted to increase (or decrease) by at least 0.5% on day $i+h$, we buy (or sell) the option. The practice of using filtering thresholds to avoid noisy signals is also used in Goncalves and Guidolin (2006).

[19] A short position in a delta-neutral strategy is essentially betting against volatility, while a long position is betting that volatility will increase (see Coval and Shumway, 2001, Bakshi and Kapadia, 2003, Driessen and Maenhout, 2007, Gao, Xing, and Zhang (2018)).

5.2 Performance of Trading Strategies

In evaluating the performance of our trading strategies, we report two key metrics: mean returns (MR) as a percentage and the annualized Sharpe ratio (SR).[20] We present the detailed performance of the short delta-neutral straddle strategy exclusively in the main text due to space constraints. However, we will provide a general overview of the performance of the remaining strategies. More detailed performance metrics for these additional strategies can be found in Appendix A.11.

Figure 5 presents the results for the short delta-neutral straddles. Consistent with the statistical results, in the shorter forecasting horizons, the classical models, such as fRW and AHBS-DNN, perform better fNTK. However, for forecasting horizons longer than one week, fNTK-backed strategies yield remarkable mean returns and Sharpe ratios. The bi-weekly mean return of 7.49% and monthly return of 14.40% represent an improvement ranging from 27% to 245%, while the Sharpe ratios ranging between 1.30 and 1.83 translate into a substantial 90% to 675% relative enhancement in trading outcomes compared to the functional random walk benchmark. For further details, such as the performance per year, consult Table A.13. While other nonlinear kernel models also produce impressive results, fNTK remains the clear leader in terms of overall performance.

Table A.13 in the Appendix displays a multi-year performance analysis revealing that during high market volatility, especially in 2020, most benchmarks had negative returns and Sharpe ratios. In contrast, fNTK consistently maintains a positive Sharpe ratio, ranging between 0.04 and 0.75. During 2019 and 2021, all nonlinear functional models yield profitable strategies, with Sharpe ratios between 0.77 and 5.37. These findings underscore the role of nonlinearities in modeling IVS dynamics and extracting valuable trading signals. Moreover, they suggest that integrating fNTK into trading strategies is particularly rewarding during turbulent market conditions.

When evaluating the performance of a trading strategy, it is essential to go beyond mere ag-

[20] We base our analysis on simple returns R_i. The excess return is calculated as $ER_i = R_i - \left(\exp\left(\frac{h \cdot r_{i,h}}{252}\right) - 1\right)$, where $r_{i,h}$ represents the annualized riskless interest rate with a time-to-maturity of h days. The Sharpe ratio is defined as $\frac{\mu_{ER}}{\sigma_{ER}}$, where $\mu_{ER} = \frac{1}{N_{traded}} \sum_{i=1}^{N_{traded}} ER_i$ and $\sigma_{ER} = \sqrt{\frac{1}{N_{traded}-1} \sum_{i=1}^{N_{traded}} (ER_i - \mu_{ER})^2}$ denote the mean and standard deviation of excess returns, and N_{traded} is the number of traded days.

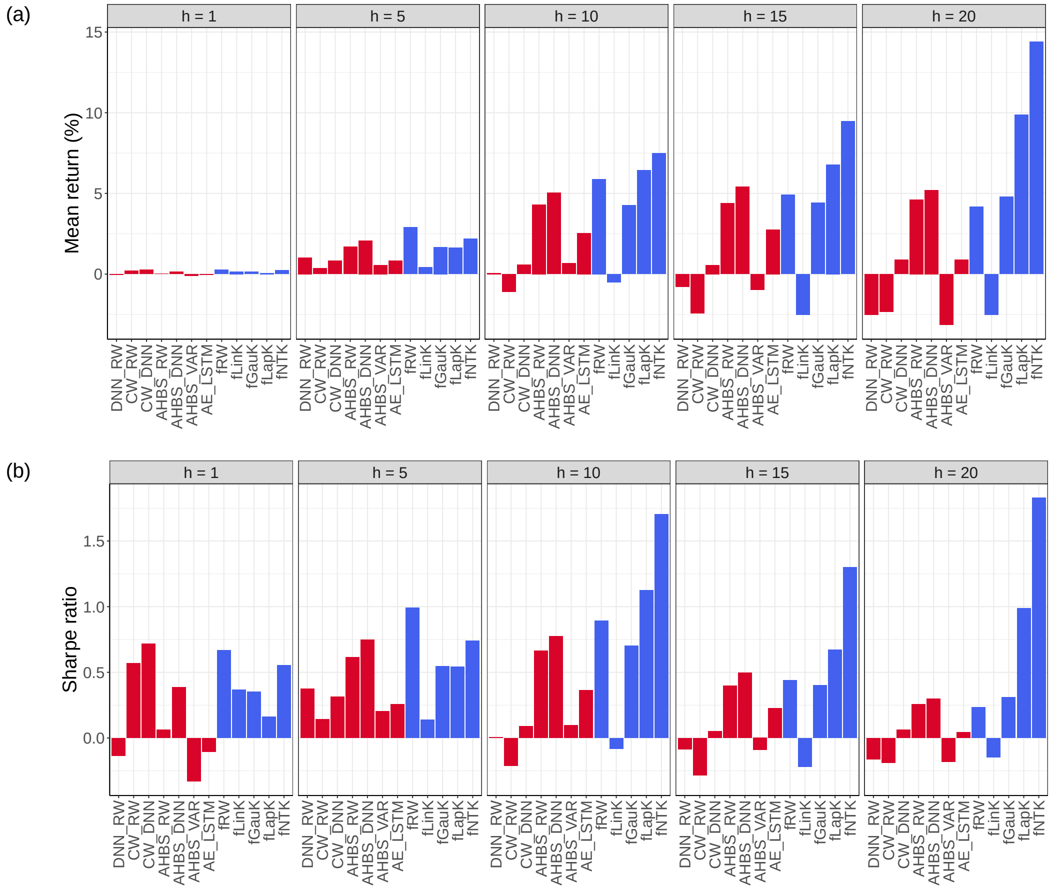


Figure 5: Mean simple returns (%) and annualized Sharpe ratio of short delta-neutral straddle strategy. The prediction period is from Jan 09, 2019 to Dec 31, 2021. The blue color is for functional models, while the red color is to indicate classical models.

gregate results. Different options, distinguished by their features, carry varying degrees of risk. Therefore, it is crucial to investigate how their predictability impacts strategy performance across different moneyness and maturity groups for a more comprehensive understanding of the strategy's effectiveness. In Figure 6, we present data on the mean daily traded volume, mean returns, and annualized Sharpe ratio for short call delta-hedging, short put delta-hedging, and short delta-neutral straddle strategies, all at the 20-step ahead forecasting horizon, representative of longer forecasting horizons. The results reveal a clear trend: trading options with higher moneyness and shorter time-to-maturity consistently leads to superior trading strategy performance. Notably, the kernel-based models, with fNTK in the lead, exhibit substantial differences from the benchmark strategies. We also show trading returns of all available options on both day i and day $i+h$. The

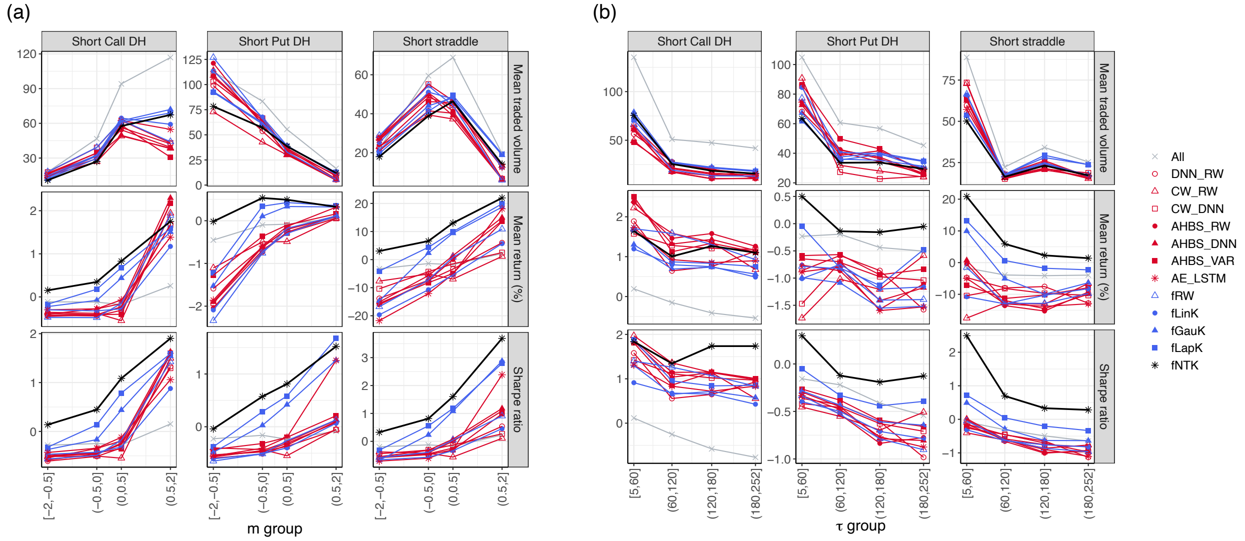


Figure 6: Mean traded volume, mean simple returns (MR) in percentage, Sharpe ratio (SR) of the models for all the short trading strategies at the forecasting horizon $h = 20$, across different moneyness m groups $[-2,-0.5], (-0.5,0], (0,0.5]$, and $(0.5,2]$, and time-to-maturity τ groups $[5,60], (60,120], (120,180]$, and $(180,252]$. The prediction period is from Jan 09, 2019 to Dec 31, 2021.

benefits of employing trading signals, particularly those generated by nonlinear kernel models, are striking across all moneyness and time-to-maturity groups.

Our analysis extends to the performance of trading strategies when transaction costs are taken into account. The results for short delta-neutral straddles, presented in the Appendix Table A.14 and Figure A.11, for effective spread levels of 50%, 75%, and 100%, highlight that fNTK maintains its leading position when considering transaction costs. While there is a decrease in performance as the effective spread level increases, this drop is not substantial. For example, compared to our results with no transaction cost, fNTK has a roughly 10% reduction in mean returns and a 12% reduction in the Sharpe ratio for portfolios with a 20-day horizon at the 100% spread. Furthermore, nonlinear models consistently outperform linear and classical models across various spread values, especially for longer forecasting horizons. Overall, our results suggest that the trading strategy results remain robust in the presence of transaction costs, with a moderate decline in performance.

Another critical variable affecting trading returns is the filtering threshold. Higher threshold values require a higher deviation from option IV on the day i for the option to be traded, resulting in fewer trades. In Appendix A.11, we repeat the trading exercises with two additional threshold values: 5% and 10%. The results in Table A.15 demonstrate that as threshold values increase, mean returns and Sharpe ratios for nonlinear models generally improve. For instance, in the short delta-neutral straddle strategy with a 20-step ahead prediction, the fNTK model exhibits an increase in mean return from 14.40% to 23.10% and an increase in Sharpe ratio from 1.83 to 2.20. In contrast, the fRW model's mean return decreases from 4.17% to -2.92%, and the corresponding Sharpe ratio decreases from 0.24 to -0.13.[21]

In summary, the fNTK model consistently demonstrates strong performance across various trading strategies and market conditions. It outperforms both functional alternatives and classical models in terms of mean simple returns and annualized Sharpe ratios, particularly for strategies involving delta-neutral straddle options. This robustness is evident across different prediction periods, even during turbulent market conditions such as the Covid-19 pandemic in 2020. Furthermore, fNTK's superiority holds when accounting for transaction costs, with its positive returns persisting and remaining relatively unaffected. The model's resilience is further underscored by its ability to maintain its advantage as filtering thresholds increase.

6 Conclusion

Our research reveals complex interactions between IVS functional responses and various lagged functional predictors. It highlights the superior predictive power of the fNTK estimator, especially for longer forecasting horizons. These results are not only statistically significant but also carry substantial economic implications. We demonstrate the empirical value of the fNTK-based NFAR model by conducting extensive simulations of trading strategies. The fNTK consistently outperforms alternative models, generating superior trading mean returns and Sharpe ratios across various market conditions. While rooted in IVS forecasting, our modeling framework has broader implications. Its generality makes it versatile for applications beyond IV forecasting, heralding a

[21] This trend can potentially be attributed to the fact that nonlinear models exhibit statistically superior MAPE and MCPDC measurements compared to fRW, fLinK, and classical models, particularly for longer horizons, as documented in Appendix A.9. Consequently, when the filtering threshold is raised, nonlinear models tend to capture options with more accurate high deviations, whereas benchmark models may select options with higher errors on day $i+h$.

new path in econometric modeling for analyzing complex data with nonlinear dynamics.

Several essential directions for future research merit mentioning. One critical issue is the development of adequate dimension-reduction techniques within the functional neural kernel regression framework. For example, in our future work, we intend to build on the research by Singer, Krivobokova, and Munk (2017) on kernel partial least squares and adapt their results to nonlinear functional regression using neural kernels. Additionally, we are exploring alternative methods, such as nonlinear sufficient dimension reduction for functions in Reproducing Kernel Hilbert Spaces (RKHS) with adaptive kernels, which would extend the work of Li and Song (2017) and Liang, Sun, and Liang (2022). Lastly, an innovative approach to achieving parsimony in a functional setting comes from Yao, Mueller, and Wang (2021), who introduced adaptive basis learning using deep neural networks. Incorporating an adaptive basis similar to theirs would represent a significant contribution to our research.

7 Acknowledgements

The authors would like to thank Cees Diks, Gianluca Finocchio, Gustavo Freire, and Eftychia Solea, as well as participants in the 2023 seminars at ETH Forschungsinstitut für Mathematik, ETH Risk Centre, UoE Seminar in Economics, QMUL Statistics and Data Science, mini-workshop at Optiver, Inaugural Summer Symposium on Artificial Intelligence for FinTech conference, COMPSTAT, STAT of ML, CFE/CSM, the 2024 FinEML online seminar, SoFiE conference, IAAE, Frontiers of Functional Data Analysis, the 8th NUS-ISM-ZIB-IIR-MODAL Workshop, and the Meeting of the German Committee for Econometrics, for their valuable feedback and support. The authors would also like to thank the Ministry of Education Singapore for their research funding under the project A-8000014-00-00. This research project is supported by the National Research Foundation Singapore and Monetary Authority Singapore under A-0003504-17-00. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation/Monetary Authority Singapore, Singapore.

References

Ackerer, D., N. Tagasovska, and T. Vatter (2020). “Deep smoothing of the implied volatility surface”. *Advances in Neural Information Processing Systems*, 33, 11552–11563.

Aït-Sahalia, Y., C. Li, and C. X. Li (2021a). “Closed-form implied volatility surfaces for stochastic volatility models with jumps”. *Journal of Econometrics*, 222.1, 364–392.

Aït-Sahalia, Y., C. Li, and C. X. Li (2021b). “Implied stochastic volatility models”. *The Review of Financial Studies*, 34.1, 394–450.

Aït-Sahalia, Y. and A. W. Lo (1998). “Nonparametric estimation of state-price densities implicit in financial asset prices”. *The Journal of Finance*, 53.2, 499–547.

Almeida, C., J. Fan, G. Freire, and F. Tang (2022). “Can a Machine Correct Option Pricing Models?” *Journal of Business & Economic Statistics*, 41.3, 995–1009.

Andersen, T. G., N. Fusari, and V. Todorov (2017). “Short-term market risks implied by weekly options”. *The Journal of Finance*, 72.3, 1335–1386.

Arora, S., S. S. Du, W. Hu, Z. Li, R. R. Salakhutdinov, and R. Wang (2019). “On exact computation with an infinitely wide neural net”. *Advances in Neural Information Processing Systems*, 32.

Audrino, F. and D. Colangelo (2010). “Semi-parametric forecasts of the implied volatility surface using regression trees”. *Statistics and Computing*, 20.4, 421–434.

Bakshi, G. and N. Kapadia (2003). “Delta-Hedged Gains and the Negative Market Volatility Risk Premium”. *The Review of Financial Studies*, 16.2, 527–566.

Bergeron, M., N. Fung, J. Hull, Z. Poulos, and A. Veneris (2022). “Variational Autoencoders: A Hands-Off Approach to Volatility”. *The Journal of Financial Data Science*, 4.2, 125–138.

Bernales, A. and M. Guidolin (2014). “Can we forecast the implied volatility surface dynamics of equity options? Predictability and economic value tests”. *Journal of Banking & Finance*, 46, 326–342.

Bernales, A. and M. Guidolin (2015). “Learning to smile: Can rational learning explain predictable dynamics in the implied volatility surface?” *Journal of Financial Markets*, 26, 1–37.

Black, F. and M. Scholes (1973). “The pricing of options and corporate liabilities”. *Journal of Political Economy*, 81.3, 637–654.

Bloch, D. A. and A. Böök (2020). “Predicting future implied volatility surface using TDBP-learning”. *Available at SSRN 3739514*.

Büchner, M. and B. Kelly (2022). “A factor model for option returns”. *Journal of Financial Economics*, 143.3, 1140–1161.

Carr, P. and L. Wu (2016). “Analyzing volatility risk and risk premium in option contracts: A new theory”. *Journal of Financial Economics*, 120.1, 1–20.

Chen, D., P. Hall, and H.-G. Müller (2011). “Single and multiple index functional regression models with nonparametric link”. *The Annals of Statistics*, 39.3, 1720–1747.

Chen, X. (2007). “Large sample sieve estimation of semi-nonparametric models”. *Handbook of Econometrics*, 6, 5549–5632.

Cho, H., Y. Goude, X. Brossat, and Q. Yao (2013). “Modeling and forecasting daily electricity load curves: a hybrid approach”. *Journal of the American Statistical Association*, 108.501, 7–21.

Cont, R. and J. Da Fonseca (2002). “Dynamics of implied volatility surfaces”. *Quantitative Finance*, 2.1, 45.

Coval, J. D. and T. Shumway (2001). “Expected option returns”. *The Journal of Finance*, 56.3, 983–1009.

Domingos, P. (2020). “Every model learned by gradient descent is approximately a kernel machine”. *arXiv preprint arXiv:2012.00152*.

Driessen, J. and P. Maenhout (2007). “An Empirical Portfolio Perspective on Option Pricing Anomalies”. *Review of Finance*, 11.4, 561–603.

Dumas, B., J. Fleming, and R. E. Whaley (1998). “Implied volatility functions: Empirical tests”. *The Journal of Finance*, 53.6, 2059–2106.

Fengler, M. R., W. K. Härdle, and E. Mammen (2007). “A semiparametric factor model for implied volatility surface dynamics”. *Journal of Financial Econometrics*, 5.2, 189–218.

Fengler, M. R., W. K. Härdle, and C. Villa (Oct. 2003). “The Dynamics of Implied Volatilities: A Common Principal Components Approach”. *Review of Derivatives Research*, 6.3, 179–202.

Fengler, M. R. and L.-Y. Hin (2015). “Semi-nonparametric estimation of the call-option price surface under strike and time-to-expiry no-arbitrage constraints”. *Journal of Econometrics*, 184.2, 242–261.

Fukumizu, K., F. R. Bach, and M. I. Jordan (2009). “Kernel dimension reduction in regression”.

Gao, C., Y. Xing, and X. Zhang (2018). “Anticipating uncertainty: straddles around earnings announcements”. *Journal of Financial and Quantitative Analysis*, 53.6, 2587–2617.

Goncalves, S. and M. Guidolin (2006). “Predictable dynamics in the S&P 500 index options implied volatility surface”. *The Journal of Business*, 79.3, 1591–1635.

Gu, S., B. Kelly, and D. Xiu (2020). “Empirical asset pricing via machine learning”. *The Review of Financial Studies*, 33.5, 2223–2273.

Härdle, W., J. Horowitz, and J.-P. Kreiss (2003). “Bootstrap methods for time series”. *International Statistical Review*, 71.2, 435–459.

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.

Jacot, A., F. Gabriel, and C. Hongler (2018). “Neural tangent kernel: Convergence and generalization in neural networks”. *Advances in Neural Information Processing Systems*, 31.

Jiang, C.-R. and J.-L. Wang (Feb. 2011). “Functional single index models for longitudinal data”. *The Annals of Statistics*, 39.1, 362–388.

Kadri, H., E. Duflos, P. Preux, S. Canu, and M. Davy (2010). “Nonlinear functional regression: a functional RKHS approach”. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, 374–380.

Klepsch, J. and C. Klüppelberg (2017). “An innovations algorithm for the prediction of functional linear processes”. *Journal of Multivariate Analysis*, 155, 252–271.

Lee, J., S. Schoenholz, J. Pennington, B. Adlam, L. Xiao, R. Novak, and J. Sohl-Dickstein (2020). “Finite versus infinite neural networks: an empirical study”. *Advances in Neural Information Processing Systems*, 33, 15156–15172.

Lee, J., L. Xiao, S. Schoenholz, Y. Bahri, R. Novak, J. Sohl-Dickstein, and J. Pennington (2019). “Wide neural networks of any depth evolve as linear models under gradient descent”. *Advances in neural information processing systems*, 32.

Li, B. and J. Song (2017). “Nonlinear sufficient dimension reduction for functional data”. *The Annals of Statistics*, 45.3, 1059–1095.

Liang, S., Y. Sun, and F. Liang (2022). “Nonlinear Sufficient Dimension Reduction with a Stochastic Neural Network”. *Advances in Neural Information Processing Systems*, 35, 27360–27373.

Muller, H.-G., Y. Wu, and F. Yao (Sept. 2013). “Continuously additive models for nonlinear functional regression”. *Biometrika*, 100.3, 607–622.

Müller, H. and F. Yao (Dec. 2008). “Functional Additive Models”. *Journal of the American Statistical Association*, 103.484, 1534–1544.

Park, B. U., E. Mammen, W. Härdle, and S. Borak (2009). “Time series modelling with semiparametric factor dynamics”. *Journal of the American Statistical Association*, 485, 284–298.

Redd, A. (2012). “A comment on the orthogonalization of B-spline basis functions and their derivatives”. *Statistics and Computing*, 22.1, 251–257.

Sang, P. and B. Li (2022). “Nonlinear function-on-function regression by RKHS”. *arXiv preprint arXiv:2207.08211*.

Singer, M., T. Krivobokova, and A. Munk (2017). “Kernel Partial Least Squares for Stationary Data”. *Journal of Machine Learning Research*, 18.123, 1–41.

Sun, Y. and Q. Wang (Feb. 2020). “Function-on-function quadratic regression models”. *Computational statistics & data analysis*, 142.106814, 106814.

Ulrich, M. and S. Walther (2020). “Option-implied information: What’s the vol surface got to do with it?” *Review of Derivatives Research*, 23.3, 323–355.

Vuletić, M. and R. Cont (2023). “VolGAN: a generative model for arbitrage-free implied volatility surfaces”.

Yao, J., J. Mueller, and J.-L. Wang (2021). “Deep Learning for Functional Data Analysis with Adaptive Basis Layers”. *International Conference on Machine Learning*. PMLR, 11898–11908.

Zhang, W., L. Li, and G. Zhang (2023). “A two-step framework for arbitrage-free prediction of the implied volatility surface”. *Quantitative Finance*, 23.1, 21–34.

A Appendix

A.1 Summary statistics

Panel A: Call options

	$5 \leq \tau \leq 60$	$60 < \tau \leq 120$	$120 < \tau \leq 180$	$180 < \tau$	Total
$m < -0.5$					
Contract (%)	11.46	3.04	1.55	1.33	17.38
Average IV	0.22	0.26	0.27	0.27	0.24
$\lvert m\rvert \leq 0.5$					
Contract (%)	19.87	7.09	3.09	3.72	33.77
Average IV	0.17	0.19	0.19	0.19	0.18
$0.5 < m$					
Contract (%)	32.59	8.12	4.03	4.11	48.85
Average IV	0.15	0.17	0.17	0.17	0.16
Total					
Contract (%)	63.92	18.25	8.68	9.15	100.00
Average IV	0.17	0.19	0.20	0.19	0.18

Panel B: Put options

	$5 \leq \tau \leq 60$	$60 < \tau \leq 120$	$120 < \tau \leq 180$	$180 < \tau$	Total
$m < -0.5$					
Contract (%)	18.75	5.84	2.49	2.75	29.83
Average IV	0.23	0.26	0.28	0.28	0.24
$\lvert m\rvert \leq 0.5$					
Contract (%)	20.51	7.21	3.09	3.63	34.45
Average IV	0.18	0.19	0.20	0.19	0.18
$0.5 < m$					
Contract (%)	23.03	6.56	3.07	3.07	35.72
Average IV	0.19	0.20	0.20	0.20	0.19
Total					
Contract (%)	62.29	19.62	8.65	9.45	100.00
Average IV	0.20	0.22	0.22	0.22	0.20

Table A.1: Summary Statistics for Implied Volatilities by maturity and moneyness. Percentage of contracts and mean of IV of call options (Panel A) and put options (Panel B), over different combinations of time-to-maturity (τ, in days) and moneyness (m) between January 1, 2009, and December 31, 2021.

A.2 Dimension reduction via sieve methods

Estimating an infinite number of coefficients using a finite sample is computationally infeasible. The sieves method (Chen, 2007) projects the infinite-dimensional process onto a finite parameter space, minimizing information loss. Specifically, we construct sieves, a sequence of subspaces $\{\Theta_s\}$ from the original infinite-dimensional space Θ, which is compact and non-decreasing with each subspace satisfying the condition $\Theta_s \subseteq \Theta_{s+1} \subseteq \cdots \subseteq \Theta$ and the union of these subspaces, $\bigcup_s \Theta_s$, is dense in Θ.

Given a dataset with n observations, the strategy is to select Θ_{K_n}, a parameter space of degree K_n, such that the loss function is well-defined in the finite-dimensional linear space:

$$\Theta_{K_n} = \left\{ f(\tau) \in L^2(\mathscr{C}) \mid f(\tau) = \sum_{k=1}^{K_n} \theta_k \phi_k(\tau), \sum_{k=1}^{K_n} k^2 \theta_k^2 \leq cK_n, \theta_k \in \mathbb{R}, \tau \in \mathscr{C} \right\},$$

where $\{\theta_k\}$ denotes the expansion coefficients for functional terms, and c is a positive constant that controls the growth rate of K_n. We consider K_n as a hyperparameter in the sieve approach and will address its selection in Section A.7. Under the sieves with degree K_n, the (approximated) projection operates within a finite parameter space for $k = 1, \ldots, K_n$.

A.3 Technical proofs

Lemma 1 (Isomorphism between Reproducing Kernel Hilbert Spaces). *Under Equations (3) and (9), it holds that*

$$\begin{aligned} k(\boldsymbol{x}_i, \boldsymbol{x}_j) &= \langle k(., \boldsymbol{x}_i), k(., \boldsymbol{x}_j) \rangle_{\mathfrak{M}_{\boldsymbol{x}}} \\ &= \langle K(., X_i), K(., X_j) \rangle_{\mathfrak{M}_X} = K(X_i, X_j). \end{aligned} \tag{A.1}$$

Then the RKHS $\mathfrak{M}_X$ nested on $\mathscr{H}_X$ is isometrically isomorphic to the RKHS $\mathfrak{M}_{\boldsymbol{x}}$ nested on $\mathscr{H}_{\boldsymbol{x}}$.

Proof. By Definition 2.4.15 of Hsing and Eubank (2015), two Hilbert spaces $\mathscr{H}_Z$ and $\mathscr{H}_z$ are isometrically isomorphic if there exists a bijective map $\mathscr{T} : \mathscr{H}_Z \to \mathscr{H}_z$ defined by $\mathscr{T}(Z) = z$ that is distance-preserving

$$\left\langle \mathrm{Z}_i, \mathrm{Z}_j \right\rangle_{\mathscr{H}_\mathrm{Z}} = \left\langle \mathscr{T}(\mathrm{Z}_i), \mathscr{T}(\mathrm{Z}_j) \right\rangle_{\mathscr{H}_z} = \left\langle z_i, z_j \right\rangle_{\mathscr{H}_z}, \quad \text{for all} \quad \mathrm{Z}_i, \mathrm{Z}_j \in \mathscr{H}_\mathrm{Z}. \tag{A.2}$$

Given the projection of X

$$X_i = \sum_{j=1}^{\infty} x_{ij} \psi_j, \quad \text{with } x_{ij} = \langle X_i, \psi_j \rangle_{\mathscr{H}_X}. \tag{A.3}$$

onto the orthonormal basis $\{\psi_\ell : \ell = 1, 2, \ldots\}$ where $\langle \psi_\ell, \psi_v \rangle_{\mathscr{H}_X} = 1$ if $\ell = v$ and 0 otherwise, it follows that

$$\begin{aligned} \langle X_i, X_j \rangle_{\mathscr{H}_X} &= \langle \sum_{\ell=1}^{\infty} x_{i\ell} \psi_\ell, \sum_{v=1}^{\infty} x_{jv} \psi_k \rangle_{\mathscr{H}_X} \\ &= \sum_{\ell=1}^{\infty} \sum_{v=1}^{\infty} x_{i\ell} x_{jv} \langle \psi_\ell, \psi_v \rangle_{\mathscr{H}_X} \\ &= \sum_{v=1}^{\infty} x_{iv} x_{jv} \\ &= \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle. \end{aligned} \tag{A.4}$$

If we define the map $\mathscr{V} : \mathscr{H}_X \to \mathbb{R}^\infty$ as $\mathscr{V} X = (\langle X, \psi_v \rangle_{\mathscr{H}_X})_{v=1,2,\ldots} = (x_v)_{v=1,2,\ldots}$, it is a bijective linear mapping, and is also distance-preserving as shown in Equation (A.4). Hence, by extension of Lemma 4.2. of Klepsch and Klüppelberg (2017), the function process $\{X_i\}$ is isometrically isomorphic to the vector process $\{x_i\}$.

For $k : \mathscr{H}_{\boldsymbol{x}} \times \mathscr{H}_{\boldsymbol{x}} \to \mathbb{R}$ that satisfies (9), by Equation (A.4), we have

$$\begin{aligned} k(\boldsymbol{x}_i, \boldsymbol{x}_j) &= \rho(\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle, \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle, \langle \boldsymbol{x}_j, \boldsymbol{x}_j \rangle) \\ &= \rho(\langle X_i, X_i \rangle_{\mathscr{H}_X}, \langle X_i, X_j \rangle_{\mathscr{H}_X}, \langle X_j, X_j \rangle_{\mathscr{H}_X}) \\ &= K(X_i, X_j). \end{aligned} \tag{A.5}$$

Let $\mathfrak{M}_{\boldsymbol{x}}$ be the RKHS generated by $k(., \boldsymbol{x})$. By the kernel property $K(X_i, X_j) = \langle K(., X_i), K(., X_j) \rangle_{\mathfrak{M}_X}$

and $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle k(., \boldsymbol{x}_i), k(., \boldsymbol{x}_j) \rangle_{\mathfrak{M}_{\boldsymbol{x}}}$, and Equation (A.5), we obtain

$$\langle K(.,X_i), K(.,X_j) \rangle_{\mathfrak{M}_X} = \langle k(., \boldsymbol{x}_i), k(., \boldsymbol{x}_j) \rangle_{\mathfrak{M}_{\boldsymbol{x}}}. \tag{A.6}$$

Thus, there must exist a bijective map $\mathscr{T} : \mathfrak{M}_X \to \mathfrak{M}_{\boldsymbol{x}}$ defined by $\mathscr{T}(K(.,X_i)) = k(., \boldsymbol{x}_i)$ that is distance-preserving between the two spaces, i.e.,

$$\langle K(.,X_i), K(.,X_j) \rangle_{\mathfrak{M}_X} = \langle \mathscr{T}(K(.,X_i)), \mathscr{T}(K(.,X_j)) \rangle_{\mathfrak{M}_{\boldsymbol{x}}} = \langle k(., \boldsymbol{x}_i), k(., \boldsymbol{x}_j) \rangle_{\mathfrak{M}_{\boldsymbol{x}}},$$

and $\mathfrak{M}_X$ is isometrically isomorphic to $\mathfrak{M}_{\boldsymbol{x}}$. □

Theorem 1 (Vector-to-vector regression). *Given the decomposition of X_i in Equations (3) and Y_i in Equation (2), under Assumptions (1) - (3) and Lemma 1, for a positive definite kernel k defined by Equation (9), if there is a covariance matrix $\Sigma_{\boldsymbol{x}\boldsymbol{x}}$ of $k(., \boldsymbol{x})$ that is diagonal, then the function-to-function regression model in Equation (6) may be represented equivalently by*

$$\beta_0 = \underset{\beta \in \mathscr{B}(\mathscr{H}_{\boldsymbol{y}}, \mathfrak{M}_{\boldsymbol{x}})}{\arg\min} \; \mathbb{E}\left[\| \boldsymbol{y}_i - \beta^* k(., x_i) \|^2 \right], \tag{A.7}$$

with solution $\beta_0 = \Sigma^{\dagger}_{\boldsymbol{x}\boldsymbol{x}} \Sigma_{\boldsymbol{x}\boldsymbol{y}}$. This leads to

$$\begin{aligned} \mathbb{E}[\boldsymbol{y}_i \mid \boldsymbol{x}_i] &= \beta_0^* k(., \boldsymbol{x}_i) \\ &= \Sigma_{\boldsymbol{y}\boldsymbol{x}} \Sigma^{\dagger}_{\boldsymbol{x}\boldsymbol{x}} k(., \boldsymbol{x}_i) \\ &= \mathbb{E}\left[\{ (\Sigma^{\dagger}_{\boldsymbol{x}\boldsymbol{x}} k(., \boldsymbol{x}_i))(\boldsymbol{x}) \} \boldsymbol{y} \right]. \end{aligned} \tag{A.8}$$

Proof. First, we show that the function-to-function regression can be equivalently written as a vector-to-function regression. By projecting Y_i onto the set of orthonormal basis functions $\varphi = (\varphi_1, \varphi_2, \dots)^T$ with $\varphi_j \in \mathscr{H}_Y$

$$Y_i = \sum_{j=1}^{\infty} y_{ij} \varphi_j, \quad \text{with } \; y_{ij} = \langle Y_i, \varphi_j \rangle_{\mathscr{H}_Y},$$

and letting the number of bases j go to infinity, there is no information loss in the expansion. Given the fixed form of φ, the coefficients y_{ij}-s explain the functional variables uniformly. Because Y_i is centered, its basis coefficient vector $\boldsymbol{y}_i = (y_{i1}, y_{i2}, ...)^T$ has zero mean. Since $B_0^* K(.,X_i) \in \mathscr{H}_Y$, we can express it in terms of $\{\varphi_j\}$, the basis functions of of $\mathscr{H}_Y$

$$\begin{aligned} \mathbb{E}[Y_i|X_i] &= B_0^* K(.,X_i) \\ &= \sum_{j=1}^{\infty} \langle \varphi_j, B_0^* K(.,X_i)\rangle_{\mathscr{H}_Y} \varphi_j \\ &= \sum_{j=1}^{\infty} \langle B_0 \varphi_j, K(.,X_i)\rangle_{\mathfrak{M}_X} \varphi_j \quad \text{by the property of adjoint operator} \\ &= \sum_{j=1}^{\infty} \langle b_{0j}, K(.,X_i)\rangle_{\mathfrak{M}_X} \varphi_j \\ &= \sum_{j=1}^{\infty} b_{0j}(X_i) \varphi_j \quad \text{by the reproducing properties of } K \end{aligned} \tag{A.9}$$

where $b_{0j} = B_0 \varphi_j \in \mathfrak{M}_X$ and $b_{0j}(X_i) \in \mathbb{R}$.

Taking conditional expectation of $Y_i = \sum_{j=1}^{\infty} y_{ij} \varphi_j$ leads to

$$\mathbb{E}[Y_i|X_i] = \sum_{j=1}^{\infty} \mathbb{E}[y_{ij}|X_i] \varphi_j. \tag{A.10}$$

Hence, predicting Y_i is equivalent to predicting $\boldsymbol{y}_i = (y_{i1}, y_{i2}, ...)^T$ given input $X = X_i$, basis functions $\{\varphi_j\}$, and kernel K. Furthermore, by (A.9), (A.10), and the orthonormality of $\{\varphi_j\}$, the conditional value of each y_{ij} for $j = 1, 2, ...$ given $X = X_i \in \mathscr{H}_X$ and kernel K is

$$\mathbb{E}[y_{ij}|X_i] = b_{0j}(X_i) = \langle B_0 \varphi_j, K(.,X_i)\rangle_{\mathfrak{M}_X}. \tag{A.11}$$

This means that the original function-to-function regression is equivalent to performing several regressions, where we predict the basis coefficient vector $\boldsymbol{y}_i$ of Y_i given the functional input X_i.

In the proof of Lemma 1 we showed that there is a bijective map $\mathscr{T} : \mathfrak{M}_X \to \mathfrak{M}_{\boldsymbol{x}}$ defined by $\mathscr{T}(K(.,X_i)) = k(.,\boldsymbol{x}_i)$ that is distance-preserving between the two spaces. In what follows, we provide a specification of such a bijective map. The choice we make below enables us to show theoretical equivalence between regressing the target variable on functions or their projection coefficients in a nonlinear kernel-based model.

We can project $K(.,X_i)$ onto the set of orthonormal basis in $\mathfrak{M}_X$. A natural choice for this is the set of eigenfunctions of Σ_{XX}.

If $\mathbb{E}(\|K(.,X_i)\|^2_{\mathfrak{M}_X}) < \infty$, then $K(.,X_i)$ admits the Karhunen-Loève decomposition

$$K(.,X_i) = \sum_{\ell=1}^{\infty} \zeta_{i\ell}\omega_\ell, \quad \text{with } \zeta_{i\ell} = \langle K(.,X_i), \omega_\ell \rangle_{\mathfrak{M}_X}, \tag{A.12}$$

where $\{\omega_\ell\}$ is the set of orthonormal eigenfunctions in $\mathfrak{M}_X$, and denote $\boldsymbol{\omega} = (\omega_1, \omega_2, ...)$.

Let's assume that $k(.,\boldsymbol{x}_i) = (\zeta_{i1}, \zeta_{i2}, \ldots)^T$. This is equivalent to a linear map $\mathscr{T}$, such that $\mathscr{T}K(.,X_i) = \big(\langle K(.,X_i), \omega_l \rangle_{\mathfrak{M}_X}\big)_{l=1,2,\ldots} = k(.,\boldsymbol{x}_i)$. Then, the covariance operator Σ_{XX} can be decomposed into

$$\begin{aligned}
\Sigma_{XX} &= \mathbb{E}[K(.,X) \otimes K(.,X)] \\
&= \mathbb{E}\Big[\Big(\sum_{\ell=1}^{\infty} \zeta_\ell \omega_\ell\Big) \otimes \Big(\sum_{j=1}^{\infty} \zeta_j \omega_j\Big)\Big] \\
&= \mathbb{E}\Big[\sum_{\ell=1}^{\infty}\sum_{j=1}^{\infty} \zeta_\ell \zeta_j \big(\omega_\ell \otimes \omega_j\big)\Big] \\
&= \boldsymbol{\omega}\mathbb{E}\Big[k(.,\boldsymbol{x}) \otimes k(.,\boldsymbol{x})\Big]\boldsymbol{\omega}^T \\
&= \boldsymbol{\omega}\Sigma_{\boldsymbol{xx}}\boldsymbol{\omega}^T,
\end{aligned} \tag{A.13}$$

where $\Sigma_{\boldsymbol{xx}} = \mathbb{E}\big(k(.,\boldsymbol{x}) \otimes k(.,\boldsymbol{x})\big) = \text{diag}\big(\mathbb{E}[\zeta_1^2], \mathbb{E}[\zeta_2^2], ...\big)$ is the covariance matrix of $k(.,\boldsymbol{x})$ and $\Sigma^\dagger_{\boldsymbol{xx}} = \text{diag}\big(\mathbb{E}[\zeta_1^2]^{-1}\mathbb{1}_{\mathbb{E}[\zeta_1^2]>0}, \mathbb{E}[\zeta_2^2]^{-1}\mathbb{1}_{\mathbb{E}[\zeta_2^2]>0}, ...\big)$ is its Moore-Penrose inverse[1].

[1] If Σ_{XX} is positive definite, the diagonal of the Moore-Penrose inverse of $\Sigma_{\boldsymbol{xx}}$ will have zero for the degenerate eigenvalues.

We can also express $\Sigma_{XX}^{\dagger}$ in terms of $\Sigma_{\boldsymbol{xx}}^{\dagger}$ and $\boldsymbol{\omega}$

$$\Sigma_{XX}^{\dagger} = \boldsymbol{\omega}\Sigma_{\boldsymbol{xx}}^{\dagger}\boldsymbol{\omega}^{T}. \tag{A.14}$$

Multiply both sides by ω_ℓ, we obtain

$$\begin{aligned} \Sigma_{XX}^{\dagger}\omega_\ell &= \boldsymbol{\omega}\Sigma_{\boldsymbol{xx}}^{\dagger}(\langle\omega_1,\omega_\ell\rangle,\langle\omega_2,\omega_\ell\rangle,\ldots)^{T} \\ &= (\omega_1\Sigma_{\boldsymbol{xx},1}^{\dagger},\omega_2\Sigma_{\boldsymbol{xx},2}^{\dagger},\cdots)e_\ell \\ &= \omega_\ell\Sigma_{\boldsymbol{xx},\ell}^{\dagger}. \end{aligned} \tag{A.15}$$

where $e_\ell = (0,0,\ldots,0,1,0,\ldots)^T$ is a vector with all elements equal 0 except for the $\ell-$th entry, which is equal to 1, and $\Sigma_{\boldsymbol{xx},\ell}^{\dagger}$ is the $\ell-$th diagonal term of $\Sigma_{\boldsymbol{xx}}^{\dagger}$.

Additionally, we also have

$$\begin{aligned} \langle\varphi_j,\Sigma_{YX}\omega_\ell\rangle_{\mathscr{H}_Y} &= \langle\varphi_j,\mathbb{E}[Y\otimes(K(.,X))]\omega_\ell\rangle_{\mathscr{H}_Y} \\ &= \langle\varphi_j,\mathbb{E}[Y\langle(K(.,X),\omega_\ell\rangle_{\mathfrak{M}_X}]\rangle_{\mathscr{H}_Y} \\ &= \langle\varphi_j,\mathbb{E}[Y\zeta_\ell]\rangle_{\mathscr{H}_Y} \\ &= \mathbb{E}[\langle\varphi_j,Y\rangle_{\mathscr{H}_Y}\zeta_\ell] \\ &= \mathbb{E}[y_j\zeta_\ell]. \end{aligned} \tag{A.16}$$

Equations (A.15) and (A.16) derived under the assumption that $\Sigma_{\boldsymbol{xx}}$ is diagonal will be useful to derive our following result and further rewrite $b_{0j}(X_i)$ of Equation (A.11) as

$$
\begin{aligned}
b_{0j}(X_i) &= \langle B_0 \varphi_j, K(.,X_i)\rangle_{\mathfrak{M}_X} \\
&= \langle B_0 \varphi_j, \sum_{\ell=1}^{\infty} \zeta_{i\ell}\omega_\ell\rangle_{\mathfrak{M}_X} \\
&= \sum_{\ell=1}^{\infty} \langle B_0 \varphi_j, \omega_\ell\rangle_{\mathfrak{M}_X}\ \zeta_{i\ell} \\
&= \sum_{\ell=1}^{\infty} \langle \varphi_j, B_0^* \omega_\ell\rangle_{\mathscr{H}_Y}\ \zeta_{i\ell} \\
&= \sum_{\ell=1}^{\infty} \langle \varphi_j, \Sigma_{YX}\Sigma_{XX}^{\dagger} \omega_\ell\rangle_{\mathscr{H}_Y}\ \zeta_{i\ell} \\
&= \sum_{\ell=1}^{\infty} \langle \varphi_j, \Sigma_{YX}\,\omega_\ell \Sigma_{\boldsymbol{x}\boldsymbol{x},\ell}^{\dagger}\rangle_{\mathscr{H}_Y}\ \zeta_{i\ell} \qquad \text{by Equation (A.15)} \\
&= \sum_{\ell=1}^{\infty} \langle \varphi_j, \Sigma_{YX}\,\omega_\ell\rangle_{\mathscr{H}_Y}\ \Sigma_{\boldsymbol{x}\boldsymbol{x},\ell}^{\dagger}\ \zeta_{i\ell} \\
&= \sum_{\ell=1}^{\infty} \mathbb{E}[y_j \zeta_\ell]\ \Sigma_{\boldsymbol{x}\boldsymbol{x},\ell}^{\dagger}\ \zeta_{i\ell} \qquad \text{by Equation (A.16)} \\
&= \mathbb{E}[y_j k(.,\boldsymbol{x})^T]\ \Sigma_{\boldsymbol{x}\boldsymbol{x}}^{\dagger}\ k(.,\boldsymbol{x}_i) \quad \text{since } k(.,\boldsymbol{x}) = (\zeta_1,\, \zeta_2, \ldots)^T \text{ and } \Sigma_{\boldsymbol{x}\boldsymbol{x}}^{\dagger} \text{ is diagonal} \\
&= \Sigma_{y_j\boldsymbol{x}}\Sigma_{\boldsymbol{x}\boldsymbol{x}}^{\dagger} k(.,\boldsymbol{x}_i),
\end{aligned}
\tag{A.17}
$$

where $\Sigma_{y_j\boldsymbol{x}} = \mathbb{E}[y_j k(.,\boldsymbol{x})^T]$. Following Equation (A.11) and by Equation (A.17), the conditional expectation of the $j-$th basis coefficient y_{ij} of Y_i given the vector of basis coefficients $\boldsymbol{x}_i$ of X_i is

$$
\begin{aligned}
\mathbb{E}[y_{ij}|\boldsymbol{x}_i] &= b_{0j}(X_i) \\
&= \Sigma_{y_j\boldsymbol{x}}\Sigma_{\boldsymbol{x}\boldsymbol{x}}^{\dagger} k(.,\boldsymbol{x}_i) \\
&= \mathbb{E}\left[\{(\Sigma_{\boldsymbol{x}\boldsymbol{x}}^{\dagger} k(.,\boldsymbol{x}_i))(\boldsymbol{x})\} y_j\right].
\end{aligned}
\tag{A.18}
$$

The set of equations in (A.18) can be written in the vector form, yielding a vector-to-vector regression, and the conditional expectation for $\boldsymbol{y}_i = (y_{i1}, y_{i2}, ...)^T$ is

$$\begin{aligned}
\mathbb{E}[\boldsymbol{y}_i|\boldsymbol{x}_i] &= \mathbb{E}\left[\{(\Sigma_{\boldsymbol{xx}}^{\dagger} k(.,\boldsymbol{x}_i))(\boldsymbol{x})\}\boldsymbol{y}\right] \\
&= \Sigma_{\boldsymbol{yx}}\Sigma_{\boldsymbol{xx}}^{\dagger} k(.,\boldsymbol{x}_i) \qquad \text{(A.19)} \\
&= \beta_0^* k(.,\boldsymbol{x}_i)
\end{aligned}$$

where linear operator $\beta_0 = \Sigma_{\boldsymbol{xx}}^{\dagger}\Sigma_{\boldsymbol{xy}}$, with the adjoint operator $\beta_0^* = \Sigma_{\boldsymbol{yx}}\Sigma_{\boldsymbol{xx}}^{\dagger}$ Hence, if there is a diagonal covariance matrix $\Sigma_{\boldsymbol{xx}}$ of $k(.,\boldsymbol{x})$, the original function-to-function regression model can be represented equivalently by the vector-to-vector regression.[2] □

A.4 Neural Tangent Kernal

Suppose we have a NN of depth L, $f(.;\boldsymbol{\theta}) : \mathbb{R}^{n_0} \to \mathbb{R}^{n_L}$ with parameters $\boldsymbol{\theta}$, where layers are indexed from 0 (input $\boldsymbol{x}$) to L (output $\boldsymbol{y}$), each layer containing $n_0, n_1, ..., n_L$ neurons. We use the NTK parameterization of Jacot, Gabriel, and Hongler (2018) for the NN

$$\begin{aligned}
\text{input layer} : \alpha^{(0)}(\boldsymbol{x};\boldsymbol{\theta}) &= \boldsymbol{x} \in \mathbb{R}^{n_0}, \\
\text{preactivation} : \tilde{\alpha}^{(\ell+1)}(\boldsymbol{x};\boldsymbol{\theta}) &= \frac{1}{\sqrt{n_\ell}} W^{(\ell)} \alpha^{(\ell)}(\boldsymbol{x};\boldsymbol{\theta}) + \eta b^{(\ell)}, \\
\text{activation} : \alpha^{(\ell+1)}(\boldsymbol{x};\boldsymbol{\theta}) &= \sigma(\tilde{\alpha}^{(\ell+1)}(\boldsymbol{x};\boldsymbol{\theta})), \\
\text{output layer} : f(\boldsymbol{x};\boldsymbol{\theta}) &= \tilde{\alpha}^{(L)}(\boldsymbol{x};\boldsymbol{\theta}) \in \mathbb{R}^{n_L},
\end{aligned}$$

where parameters $\boldsymbol{\theta}$ consist of the connection matrices $W^{(\ell)} \in \mathbb{R}^{n_{\ell+1} \times n_\ell}$, and bias vectors $b^{(\ell)} \in \mathbb{R}^{n_{\ell+1}}$. All parameters are initialized as i.i.d. Gaussians $\mathcal{N}(0,1)$; the constant $\eta > 0$ controls the influence of the bias on the training; and the nonlinear activation function $\sigma(.)$ is applied element-wise, to each element of $\tilde{\alpha}^{(\ell+1)}(\boldsymbol{x};\boldsymbol{\theta})$. Given a training dataset $\{(\boldsymbol{x}_i, \boldsymbol{y}_i) : i = 1, ..., n\}$, the least-squares loss is defined as

[2] Note that this diagonality assumption for $\Sigma_{\boldsymbol{xx}}$ is novel and has not been used in the existing literature. It provides a sufficient (though not necessary) condition for establishing the equivalence between nonlinear kernel function-on-function regression and vector-on-vector regression.

$$\mathscr{L}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \|\boldsymbol{y}_i - f(\boldsymbol{x}_i;\boldsymbol{\theta})\|^2. \tag{A.20}$$

The parameters of the NN model are updated and learned by minimizing a least-squares loss function $\mathscr{L}(\boldsymbol{\theta})$ via the back-propagation algorithm and the Gradient Descent (GD) method with learning rate ζ. At each updating step s, the parameters $\boldsymbol{\theta}_s$ are updated to $\boldsymbol{\theta}_{s+1}$ by

$$\boldsymbol{\theta}_{s+1} = \boldsymbol{\theta}_s - \zeta \nabla_{\boldsymbol{\theta}} \mathscr{L}(\boldsymbol{\theta}_s), \tag{A.21}$$

with $\nabla_{\boldsymbol{\theta}} \mathscr{L}(\boldsymbol{\theta}_s)$ the gradient of the loss function with respect to $\boldsymbol{\theta}$ at step s. Using GD, the weights and biases in all layers are simultaneously updated towards the parameters that minimize the loss function, effectively training the neural network to make accurate predictions. Let the prediction vector of the NN be $f(\boldsymbol{x};\boldsymbol{\theta}) = (f_1(\boldsymbol{x};\boldsymbol{\theta}),...,f_{n_L}(\boldsymbol{x};\boldsymbol{\theta}))^T$, with $f_j(\boldsymbol{x};\boldsymbol{\theta})$ denoting the j-th output of the NN. Daniely, Frostig, and Singer (2016) have shown that at initialization, as the number of neurons in each layer goes to infinite, the outputs $f_j(\boldsymbol{x};\boldsymbol{\theta})$ for $j = 1,...,n_L$ tend to i.i.d. centered Gaussian processes (GP) with scalar covariance $\Sigma^{(L)}$ defined recursively by

$$\begin{aligned} \Sigma^{(1)}(\boldsymbol{x},\boldsymbol{x}') &= \frac{1}{n_0}\boldsymbol{x}^T\boldsymbol{x}' + \eta^2 \\ \Sigma^{(\ell+1)}(\boldsymbol{x},\boldsymbol{x}') &= \mathbb{E}_{\boldsymbol{\theta}}\Big[\sigma(f(\boldsymbol{x}))\sigma(f(\boldsymbol{x}'))\Big] + \eta^2, \end{aligned} \tag{A.22}$$

where $\Sigma^{(1)}(\boldsymbol{x},\boldsymbol{x}')$ can be computed from the network architecture. Suppose the NN is trained with gradient descent over a number of updating steps indexed by s. Thus, $f(\boldsymbol{x};\boldsymbol{\theta}_s)$ is the output of the NN using $\boldsymbol{\theta} = \boldsymbol{\theta}_s$, and $\nabla_{\boldsymbol{\theta}} f(\boldsymbol{x};\boldsymbol{\theta}_s)$ is the rate of change in output $f(\boldsymbol{x};\boldsymbol{\theta}_s)$ with respect to parameter $\boldsymbol{\theta}$ at step s. The empirical neural tangent kernel matrix in $\mathbb{R}^{n_L \times n_L}$ for the depth L NN is defined as

$$\tilde{k}_s^{(L)}(\boldsymbol{x},\boldsymbol{x}') = \nabla_{\boldsymbol{\theta}}^T f(\boldsymbol{x};\boldsymbol{\theta}_s) \nabla_{\boldsymbol{\theta}} f(\boldsymbol{x}';\boldsymbol{\theta}_s). \tag{A.23}$$

By Theorem 1 and Theorem 2 in Jacot, Gabriel, and Hongler (2018), for $\sigma(.)$ being Lipschitz, as $n_1, n_2, \ldots, n_{L-1} \to \infty$, the empirical NTK $\tilde{k}_s^{(L)}(\boldsymbol{x}, \boldsymbol{x}')$ converges in probability to a deterministic limiting kernel matrix $k_\infty^{(L)}(\boldsymbol{x}, \boldsymbol{x}') \otimes I_{n_L}$ for all $\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{R}^{n_0}$ and $s \geq 0$, where I_{n_L} is an identity matrix in $\mathbb{R}^{n_L \times n_L}$. This implies that an NN with n_L outputs behaves asymptotically like n_L NNs with scalar outputs trained independently. The scalar kernel $k_\infty^{(L)}(\boldsymbol{x}, \boldsymbol{x}') : \mathbb{R}^{n_0} \times \mathbb{R}^{n_0} \to \mathbb{R}$ is defined recursively

$$\begin{aligned} k_\infty^{(1)}(\boldsymbol{x}, \boldsymbol{x}') &= \Sigma^{(1)}(\boldsymbol{x}, \boldsymbol{x}') \\ k_\infty^{(\ell+1)}(\boldsymbol{x}, \boldsymbol{x}') &= k_\infty^{(\ell)}(\boldsymbol{x}, \boldsymbol{x}')\dot{\Sigma}^{(\ell+1)}(\boldsymbol{x}, \boldsymbol{x}') + \Sigma^{(\ell+1)}(\boldsymbol{x}, \boldsymbol{x}'), \end{aligned} \tag{A.24}$$

where $\dot{\Sigma}^{(\ell+1)}(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}_{\boldsymbol{\theta}}\left[\dot{\sigma}(f(\boldsymbol{x}))\dot{\sigma}(f(\boldsymbol{x}'))\right]$ and $\dot{\sigma}$ is the derivative of σ with respect to $\boldsymbol{\theta}$. This means that the empirical NTK is independent of the initialization value and is solely determined by the NN architecture. Training the NN under least-square loss $\mathscr{L}(\boldsymbol{\theta})$ by gradient descent is equivalent to a kernel regression using the NTK. Although the limiting NTK $k_\infty^{(L)}$ has a different form compared to the standard kernels, it is suitable for our modeling framework. It is defined recursively via the covariance of the GPs, $\Sigma^{(\ell)}$ in (A.22) and (A.24) and utilizes the inner product of $\boldsymbol{x}$ and $\boldsymbol{x}'$ in each iterative steps. Hence, it satisfies Equation (9) and can be used in Equation (12). Note that in practice, it is infeasible to implement an NN with infinite widths. However, for large enough widths of the hidden layers, the NN provides a fair approximation of $k_\infty^{(L)}$.

A.5 Simulation study

In this section, we study the finite-sample performance of the NFAR estimator. We consider linear and nonlinear dynamics of the coefficients in a Ad-hoc Black–Scholes (AHBS) model, with additional higher orders of moneyness m and time-to-maturity τ (Dumas, Fleming, and Whaley, 1998).

For each day i, for a given moneyness and time-to-maturity (TTM), the implied volatility surface $IV(.,.)$ is simulated at discrete points $(m_{j,i}, \tau_{j,i})$

$$\begin{aligned} IV(m_{j,i}, \tau_{j,i}) =& \alpha_{0,i} + \alpha_{1,i} m_{j,i} + \alpha_{2,i} m_{j,i}^2 + \alpha_{3,i} \tau_{j,i} + \alpha_{4,i} \tau_{j,i}^2 + \alpha_{5,i} m_{j,i} \tau_{j,i} \\ & + \alpha_{6,t} m_{j,i}^3 + \alpha_{7,t} \tau_{j,i}^3 + \alpha_{8,t} m_{j,i}^2 \tau_{j,i}^3 + \alpha_{9,t} m_{j,i}^3 \tau_{j,i}^2 + \varepsilon_{j,i} \end{aligned}$$

for $i = 1,2,...,n$ and $j = 1,2,...,J_i$, where $IV(m_{j,i}, \tau_{j,i})$, $m_{j,i}$ and $\tau_{j,i}$ are the simulated implied volatilities, moneyness, and TTM (in years) of the option j on day i, respectively; $\varepsilon_{j,i}$ is an i.i.d normally distributed random error term with mean 0 and standard deviation 0.01, and J_i is the number of options available for day i. For simplicity, we assume that the IVS are observed at regularly spaced points on each day i. The moneyness m values are set at 50 equidistant points in the range of -2.5 to 2.5, while for the time-to-maturity τ, we use the sequence of values from 0.02 to 1 by step of 0.05. This mimics our real data setup where we focus on options with $m \in [-2.5, 2.5]$ and time-to-maturity τ of at least 5 trading days and at most one year.

We set $n = 2000$, that is, the sample size, which is split into $n_1 = [1, 1200]$ as the training set to perform in-sample estimation, $n_2 = [1201, 1600]$ as the validation set for tuning hyperparameters, and $n_3 = [1601, 2000]$ as the test set to conduct the out-of-sample prediction. The temporal dependence between IVS is captured by the simulated dynamics of $\alpha = (\alpha_0, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5)^T$. We consider the two experiments described below. The generation was repeated 100 times in each experiment.

In the first experiment, we aim to investigate the performance of functional models under simple *linear* dynamics. To reflect the values of observed implied volatility of S&P 500 options, each α_ℓ, $\ell = 0, 1, ..., 5$ is simulated with an autoregressive (AR) model

$$\alpha_{\ell,i} = a_\ell \alpha_{\ell,i-1} + e_{\ell,i},$$

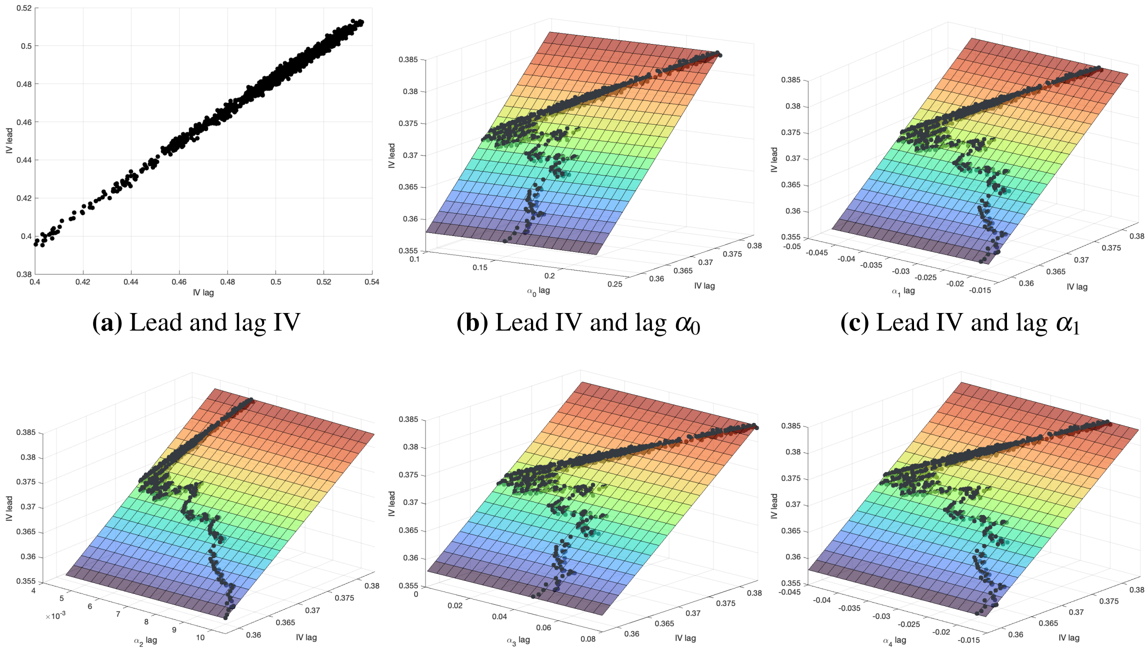
with unconditional mean and variance that match the mean and variance of α_ℓ, $\ell = 0, 1, ..., 5$ estimated from our S&P 500 data in the sample period 2009 to 2021. The other parameters, α_ℓ, $\ell = 6, 7, ..., 9$, are simulated by the interactions of $\{\alpha_\ell\}_{\ell=0}^5$: $\alpha_{6,i} = \alpha_{1,i} + \alpha_{2,i}$; $\alpha_{7,i} = \alpha_{2,i} + \alpha_{3,i}$; $\alpha_{8,i} =$

$\alpha_{0,i} + \alpha_{1,i} \times \alpha_{2,i}$ and $\alpha_{9,i} = \alpha_{2,i} + \alpha_{3,i} \times \alpha_{4,i}$. The error term $e_{\ell,i}$ is sampled independently from $\mathcal{N}(0, \sigma_\ell^2)$ where σ_ℓ^2 is set to be 2% of the variance of α_ℓ estimated from the S&P 500 data.

Next, we have the *nonlinear* experiment, in which we incorporate the nonlinearity of temporal dependence of IVS by using a nonlinear model to simulate the dynamics of each α_ℓ, $\ell = 0, 1, ..., 5$

$$\alpha_{\ell,i} = 2\sin(\alpha_{\ell,i-1}) + 4\cos(\alpha_{\ell,i-1}) + u_{\ell,i}.$$

The simulated values of α_ℓ, $\ell = 0, 1, ..., 5$ are rescaled to be in the range estimated from real data. The other parameters, α_ℓ, $\ell = 6, 7, ..., 9$, are simulated by the interactions of $\{\alpha_\ell\}_{\ell=0}^5$: $\alpha_{6,i} = \alpha_{1,i} + \alpha_{2,i}$; $\alpha_{7,i} = \alpha_{2,i} + \alpha_{3,i}$; $\alpha_{8,i} = \alpha_{0,i} + \alpha_{1,i} \times \alpha_{2,i}$ and $\alpha_{9,i} = \alpha_{2,i} + \alpha_{3,i} \times \alpha_{4,i}$. The error term $u_{\ell,i}$ is sampled independently from $\mathcal{N}(0, \sigma_\ell^2)$ where σ_ℓ^2 is set to be 2% of the variance of α_ℓ estimated from the S&P 500 data.



(a) Lead and lag IV

(b) Lead IV and lag α_0

(c) Lead IV and lag α_1

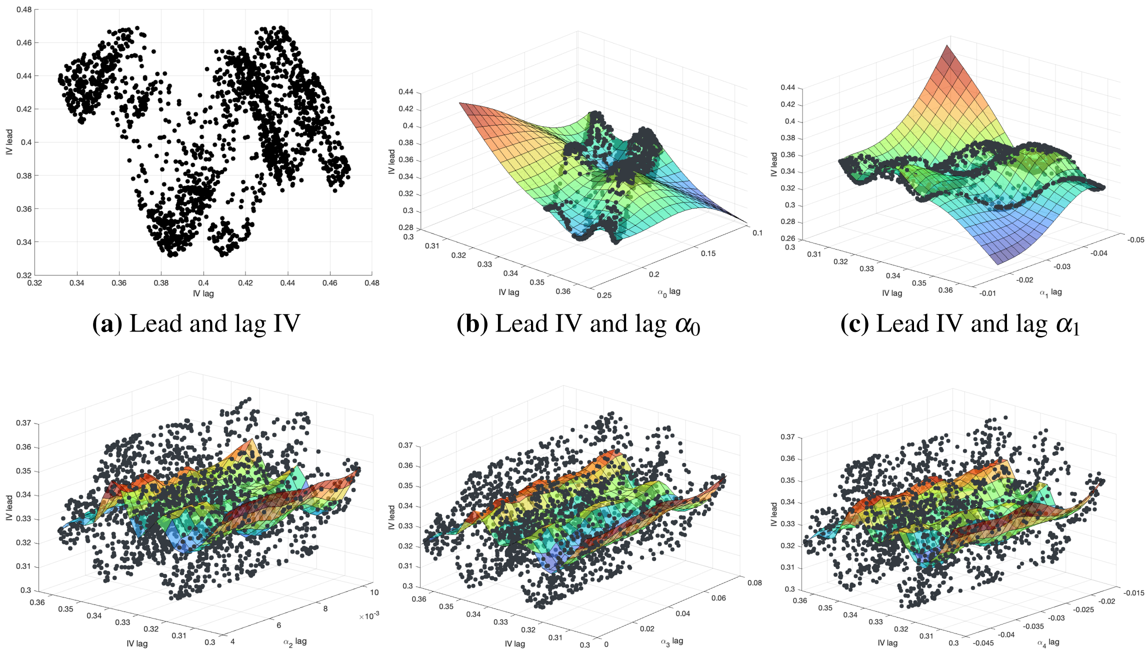
(d) Lead IV and lag α_2

(e) Lead IV and lag α_3

(f) Lead IV and lag α_4

Figure A.1: Lead values of $IV(0, 0.5)$ on day $i+1$ versus its lag values on day i in (a) and each $\alpha_\ell, \ell = 0, ..., 4$ on day i in (b) to (f), in one of the simulations for the linear experiment.

Figure A.1 (a) shows the linear relationships between lead and lag of $IV(0, 0.5)$, i.e., IV at moneyness $m = 0$ and time-to-maturity of half a year, time series, and (b) to (f) display the 3D



(a) Lead and lag IV **(b)** Lead IV and lag α_0 **(c)** Lead IV and lag α_1

(d) Lead IV and lag α_2 **(e)** Lead IV and lag α_3 **(f)** Lead IV and lag α_4

Figure A.2: Lead values of $IV(0,0.5)$ on day $i+1$ versus its lag values on day i in (a) and each $\alpha_\ell, \ell=0,\ldots,4$ on day i in (b) to (f), in one of the simulations for the nonlinear experiment.

plots when we incorporate in the lag of each parameter $\alpha_\ell, \ell=0,\ldots,4$. In Figure A.2, we illustrate IV of day $i+1$ as a nonlinear function of the parameters $\alpha_{\ell,i}, \ell=0,\ldots,4$. The sine-cosine dynamics are well reflected in the lead-lag IV relationship in Figure A.2 (a), and also in the 3D plots with the lag of each parameter α_ℓ in Figures A.2 (b) to (f).

	Linear			Nonlinear		
	RMSE	MAPE	OoR^2	RMSE	MAPE	OoR^2
fRW	2.16 (0.28)	5.07 (0.95)	95.10 (0.98)	4.91 (0.10)	10.94 (0.24)	64.36 (1.30)
fLinK	2.16 (0.28)	5.07 (0.95)	95.10 (0.98)	3.65 (0.06)	8.23 (0.13)	80.35 (0.55)
fGauK	2.19 (0.30)	5.12 (0.98)	94.95 (1.18)	2.79 (0.04)	6.49 (0.09)	88.49 (0.27)
fLapK	2.19 (0.30)	5.12 (0.99)	94.94 (1.20)	2.47 (0.04)	5.84 (0.09)	90.98 (0.27)
fNTK1	2.16 (0.28)	5.07 (0.95)	95.10 (0.98)	2.00 (0.02)	4.92 (0.07)	94.09 (0.11)
fNTK3	2.16 (0.28)	5.07 (0.95)	95.09 (0.98)	1.83 (0.02)	4.58 (0.07)	95.04 (0.08)
fNTK5	2.16 (0.28)	5.07 (0.95)	95.09 (0.98)	1.84 (0.03)	4.60 (0.07)	94.99 (0.12)

Table A.2: Mean and standard deviation (in brackets) of prediction accuracy in terms of RMSE, MAPE and OoR^2 of the functional models over the test set T_3, under Linear and Nonlinear experiments.

Table A.2 shows the mean and standard deviation of prediction performances in terms of RMSE, MAPE, and OoR^2 for the test set n_3 in the linear and nonlinear experiments. As expected, we observe that for the linear experiment, the linear model is sufficient to capture the relationship between lead and lag IVS, with 2.16% RMSE and 95.10% OoR^2, with highly similar performance in fRW, fNTK1, fNTK2, and fNTK3 models. After introducing nonlinearity, the nonlinear models start to outperform fLinK and fRW significantly. In the nonlinear experiment, fNTK3 achieves the best performance of 1.83% RMSE, compared to 3.65% for fLinK and 4.91% for fRW. The outperformance is even more profound when we look at OoR^2 and MAPE, with fNTK achieving the best OoR^2 of 95.04%, while it is 64.36% for fRW. The other nonlinear models also perform well under the nonlinear setup, for example, RMSE is 2.47% for fLapK and 2.79% for fGauK.

A.6 Estimation with parametric kernels

Each task j can be reformulated as a regression that can be solved by estimating a kernel-ridge model. Suppose the reproducing kernel k has an eigen-expansion

$$k(\boldsymbol{x},\boldsymbol{x}') = \sum_{i=1}^{\infty} \nu_i \phi_i(\boldsymbol{x})\phi_i(\boldsymbol{x}') \tag{A.25}$$

with $\nu_i \geq 0, \sum_{i=1}^{\infty} \nu_i^2 < \infty$. We can express each function $f_j \in \mathfrak{M}_{\boldsymbol{x}}$ in terms of these eigenfunctions

$$f_j(\boldsymbol{x}) = \sum_{\ell=1}^{\infty} c_{j\ell}\phi_\ell(\boldsymbol{x}) \tag{A.26}$$

with a generalized ridge penalty

$$J(f_j) = \|f_j\|^2_{\mathfrak{M}_{\boldsymbol{x}}} \overset{\text{def}}{=} \sum_{\ell=1}^{\infty} c_{j\ell}^2/\nu_\ell < \infty$$

where $\|f_j\|^2_{\mathfrak{M}_{\boldsymbol{x}}}$ is the norm induced by k. The function $f_j \in \mathfrak{M}_{\boldsymbol{x}}$ can be found by the minimization problem

$$\min_{f_j \in \mathfrak{M}_{\boldsymbol{x}}} \left[\sum_{i=1}^{n} \left(y_{ij} - f_j(\boldsymbol{x}_i)\right)^2 + \lambda \|f_j\|^2_{\mathfrak{M}_{\boldsymbol{x}}} \right] \tag{A.27}$$

By the representer theorem from Schölkopf, Herbrich, and Smola (2001), the solution of (A.27) is finite-dimensional and has the form

$$f_j(\boldsymbol{x}) = \sum_{i=1}^{n} \alpha_i k(\boldsymbol{x}, \boldsymbol{x}_i) \tag{A.28}$$

By the reproducing properties, we have $\langle k(.,\boldsymbol{x}_i), h\rangle_{\mathfrak{M}_{\boldsymbol{x}}} = h(\boldsymbol{x}_i)$ and $\langle k(.,\boldsymbol{x}_i), k(.,\boldsymbol{x}_j)\rangle_{\mathfrak{M}_{\boldsymbol{x}}} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, and the penalty term $J(f_j)$ can be further expressed as

$$J(f_j) = \sum_{i=1}^{n} \sum_{\ell=1}^{n} k(\boldsymbol{x}_i, \boldsymbol{x}_\ell)\alpha_i \alpha_\ell \tag{A.29}$$

Let $\tilde{\boldsymbol{y}}_j = (y_{1j}, \ldots, y_{nj})^T$, $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_n)^T$, and the gram matrix G where $G_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$. From (A.27), (A.28) and (A.29), we can rewrite the minimization problem in terms of $\boldsymbol{\alpha}$ and G

$$\min_{\boldsymbol{\alpha}} \quad (\tilde{\boldsymbol{y}}_j - G\boldsymbol{\alpha})^T (\tilde{\boldsymbol{y}}_j - G\boldsymbol{\alpha}) + \lambda \boldsymbol{\alpha}^T G \boldsymbol{\alpha}. \tag{A.30}$$

Thus, the solution for $\boldsymbol{\alpha}$ is simply $\widehat{\boldsymbol{\alpha}} = (G + \lambda \boldsymbol{I})^{-1} \tilde{\boldsymbol{y}}_j$. The solution for the kernel ridge regression is therefore $\hat{h}_j(\boldsymbol{x}) = \sum_{i=1}^{n} \widehat{\alpha}_i k(\boldsymbol{x}_i, \boldsymbol{x})$. The minimization problem in (A.30) is solved in one step for known kernels.

A.7 Hyperparameters

There are several hyperparameters involved in the estimation process. The sieve hyperparameter K_n dictates the dimensions of the reduced parameter space. Tuning hyperparameters γ and λ control the bandwidth and sparsity of curve predictors in regressions with parametric kernels. Additionally, for the NTK, unique hyperparameters pertain to the NN architecture.

Sieve approximation hyperparameters. The number of sieves for the predictor and response curves depends on the degrees of the orthogonal splines. For simplicity, the degrees of splines in the time-to-maturity and moneyness directions are assumed to be the same. We experimented different degrees 2, 3, 4, 5 and 6, which correspond to 9, 16, 25, 36 and 49 total number of basis functions, respectively. The results are reported in Table A.4.

Ridge regularization and bandwidth hyperparameters. Kernel ridge regression is implemented with a ridge regularization strength $\lambda > 0$ for each kernel. The hyperparameter λ modulates the balance between fitting the training data and preventing overfitting, dictating the extent to which the regularization term influences the final solution. The hyperparameters γ and λ are fine-tuned across a value grid, ranging from 0.005 to 0.025 for γ and from 10^{-5} to 10^{-1} for λ, at the start of every six months using cross-validation (see, e.g., Yao, Müller, and Wang (2005)).

NN hyperparameters. Implementation of an NN model requires the specification of several hyperparameters, including the number of hidden layers, choice of activation functions, and regularization strengths. However, an exhaustive search for the optimal architecture by evaluating infinite hyperparameter combinations is typically impractical. In this study, we employ an NN architecture featuring large widths of 500 neurons in each hidden layer. This design ensures that the empirical NTK of the NN closely approximates its limiting NTK. We opt for three hidden layers and deploy the ReLU activation function across all of them, primarily since most theoretical results for the NTK align with ReLU. We also conduct a robustness test using NNs with one and five hidden layers, with findings presented in Appendix A.9.

The NN parameters are refined and learned by minimizing $\mathscr{L}(\mathbf{\Omega}_{MLP^{(L)}})$ through the gradient descent optimizer with a learning rate of 0.05. Every six months, we adjust the weight decay rate over a set of five possible values, spanning from 10^{-5} to 10^{-1}. Weight decay, also recognized as L2 regularization, offers multiple benefits: it counteracts overfitting, enhances generalization to unseen datasets, regulates model complexity, lowers sensitivity to noise, and stabilizes the training process. Through penalization of large weight values, weight decay promotes the development of simpler models, reducing susceptibility to overfitting and ensuring better generalization to novel data.

For kernel ridge regression and neural network models, we periodically update hyperparameters every six months and adjust parameters daily for out-of-sample forecasting. Initially, our model is trained on data from the past 2500 days, divided into a 2000-day training set and a 500-

day validation set. We select the hyperparameters that minimize the RMSE on the validation set, and this value is then used as hyperparameters. These hyperparameters are applied daily to update the model using a moving window of training data over the following six months. At the end of this period, we reassess and adjust the hyperparameters as necessary.

A.8 Alternative models

A.8.1 Random Walk with Deep Neural Network model

For day i with observed values $IV_i(\tau_j, m_j), j = 1, ..., n_i$, we use a fully connected neural network $g(.)$ with three hidden layers with 32, 16, and 8 neurons, respectively to best approximate the curvatures of the implied volatility surface, by minimizing

$$\frac{1}{n_i} \sum_{j=1}^{n_i} \left[g(\tau_j, m_j) - IV_i(\tau_j, m_j) \right]^2,$$

In an h-step ahead forecasting, the IVS predicted by the DNN-RW model for day $i+h$ is given by $\hat{g}(\tau_j, m_j), j = 1, ..., n_{i+h}$. As shown in Almeida et al. (2022), the following minimization problems are equivalent:

$$\frac{1}{n_i} \sum_{j=1}^{n_i} \left[g(\tau_j, m_j) - IV_i(\tau_j, m_j) \right]^2 = \frac{1}{n_i} \sum_{j=1}^{n_i} \left[g(\tau_j, m_j) - c - IV_i(\tau_j, m_j) \right]^2$$

where c is any constant. This implies that a direct nonparametric fit to the implied volatility surface can be seen as a correction of the Black-Scholes model, which predicts a flat surface $IV_i(\tau_j, m_j) = c \quad \forall j = 1, ..., n_i$.

A.8.2 Car and Wu model

Carr and Wu (2016) proposes an option pricing framework that models the near-term dynamics of the implied volatility across different strikes and expiries. For an option with strike K and time to maturity τ, the dynamics of the underlying spot price S_i and the option implied volatility $IV_i(K, \tau)$

under the risk-neutral measures are captured as

$$\begin{aligned} dS_i/S_i &= \sqrt{v_i}dW_i, \\ dIV_i(K,\tau)/IV_i(K,\tau) &= e^{-\eta_i\tau}(a_i dt + w_i dZ_i), \end{aligned} \tag{A.31}$$

where v_i denotes the time-i instantaneous variance rate of the underlying asset log-returns, a_i is the average drift of the implied volatility, and the exponential dampening parameter $e^{-\eta_i\tau}$ accommodates the empirical observation that implied volatilities of long-dated options tend to move less. W_i and Z_i are the Wiener processes with correlation process $\rho_i \in [-1,1]$. Additionally, a_i, w_i and η_i are stochastic processes independent of K, τ and $IV_i(K,\tau)$.

Denote $k = log(KS_i)$ the relative strike. It is shown by Carr and Wu (2016) that the square implied volatility $IV_i^2(k,\tau)$ satisfies the quadratic equation

$$\frac{1}{4}e^{-2\eta_i\tau}w_i^2\tau^2 IV_i^4 + (1 - 2e^{-\eta_i\tau}a_i\tau - e^{-\eta_i\tau}w_i\rho_i\sqrt{v_i}\tau)IV_i^2 - (v_i + 2e^{-\eta_i\tau}w_i\rho_i\sqrt{v_i}k + e^{-2\eta_i\tau}w_i^2k^2) = 0. \tag{A.32}$$

The implied volatility surface on day i is fitted with the values of parameters $\boldsymbol{\theta}_i = (v_i, a_i, w_i, \eta_i, \rho_i)$ at i. Given the set of options on day i with implied volatility $IV_{j,i}$, relative strike $k_{j,i}$ and maturity $\tau_{j,i}$, the parameters $\boldsymbol{\theta}_i$ are estimated by minimizing the nonlinear least squares

$$\begin{aligned} \hat{\boldsymbol{\theta}}_i = \arg\min_{\boldsymbol{\theta}_i} \sum_{i=1}^{n_i} \Big[&\frac{1}{4}e^{-2\eta_i\tau_{j,i}}w_i^2\tau_{j,i}^2 IV_{j,i}^4 + (1 - 2e^{-\eta_i\tau_{j,i}}a_i\tau_{j,i} - e^{-\eta_i\tau_{j,i}}w_i\rho_i\sqrt{v_i}\tau_{j,i})IV_{j,i}^2 \\ &- (v_i + 2e^{-\eta_i\tau_{j,i}}w_i\rho_i\sqrt{v_i}k_{j,i} + e^{-2\eta_i\tau_{j,i}}w_i^2k_{j,i}^2)\Big]. \end{aligned} \tag{A.33}$$

Carr and Wu Random Walk (CW-RW). With the estimated parameters $\hat{\boldsymbol{\theta}}_i$, the parameters on day $i+h$ are predicted to be $\hat{\boldsymbol{\theta}}_{i+h} = \hat{\boldsymbol{\theta}}_i$ and hence the implied volatility $IV_{i,i+h}$ of day $i+h$ predicted by the CW-RW model is obtained by solving equation (A.32) using $\hat{\boldsymbol{\theta}}_{i+h}$ as inputs as well as the option relative strike and time to maturity.[3]

Carr and Wu with Deep Neural Network (CW-DNN). Following Almeida et al. (2022), given

[3] We are grateful to Gustavo Freire for sharing his codes implementing the Carr and Wu model.

the fitted parametric Carr and Wu model, we train a feedforward neural network $g(.)$ on the model-implied pricing errors to correct for mispricing and boost performance. Using a set of n_i options observed on day i, we first fit the CW model to the observed values $IV_{j,i} = IV_i(\tau_j, k_j), j = 1, ..., n_i$, obtaining $\widehat{IV}_{j,i}$ as fitted values and $\hat{\varepsilon}_{CW,j,i} = IV_{j,i} - \widehat{IV}_{j,i}$ as model implied pricing errors. Then, we estimate the pricing error surface $\varepsilon_{CW,j,i}$ nonparametrically by minimizing

$$\frac{1}{n_i}\sum_{i=1}^{n_i}\left[g(\tau_i, k_i) - \hat{\varepsilon}_{CW,j,i}\right]^2.$$

We follow the same setup as in Almeida et al. (2022), where $g(.)$ is a standard fully connected neural network with three hidden layers with 32, 16, and 8 neurons, respectively. The function $\hat{g}(\tau, k)$ is learned to best approximate the pricing error surface. Thus, in an $h-$step ahead forecasting, the IVS predicted by the CW-DNN model is given by the sum of the projected value of the CW model and its neural network correction: $\widehat{IV}_{i+h}(\tau_j, m_j) + \hat{g}(\tau_j, m_j), j = 1, ..., n_{i+h}$.

A.8.3 Ad-hoc Black–Scholes model

For each day i, the AHBS model is estimated with a cross-section of $j = 1, ..., n_i$ options using the following regression

$$IV_{j,i} = \alpha_{0,i} + \alpha_{1,i} m_{j,i} + \alpha_{2,i} m_{j,i}^2 + \alpha_{3,i}\tau_{j,i} + \alpha_{4,i}\tau_{j,i}^2 + \alpha_{5,i} m_{j,i}\tau_{j,i} + \varepsilon_{j,i}, \tag{A.34}$$

where $IV_{j,i}$, $m_{j,i}$ and $\tau_{j,i}$ are the observed implied volatilities, moneyness, and time to maturity (in years) of the option j on day i, respectively; $\varepsilon_{j,i}$ is the random error term, and n_i is the number of options available for day i.

Ad-hoc Black–Scholes Random Walk (AHBS-RW). With the estimated parameters $\hat{\boldsymbol{\alpha}}_i = (\hat{\alpha}_{0,i}, \hat{\alpha}_{1,i}, \hat{\alpha}_{2,i}, \hat{\alpha}_{3,i}, \hat{\alpha}_{4,i}, \hat{\alpha}_{5,i})^T$, the AHBS parameters on day $i+h$ is predicted to be $\hat{\boldsymbol{\alpha}}_{i+h} = \hat{\boldsymbol{\alpha}}_i$. The implied volatility $IV_{i,i+h}$ forecasted by the AHBS-RW model is obtained by Equation (A.34) and $\hat{\boldsymbol{\alpha}}_{i+h}$.

Ad-hoc Black–Scholes with Deep Neural Network (AHBS-DNN). First, the AHBS model to the observed values $IV_{j,i} = IV_j(\tau_j, m_j), j = 1, ..., n_i$, obtaining $\widehat{IV}_{j,i}$ as fitted values. We then define the implied pricing errors for the AHBS model as $\hat{\varepsilon}_{AHBS,j,i} = IV_{j,i} - \widehat{IV}_{j,i}$. Similar to the CW-DNN model, we use a fully connected neural network $g(.)$ with three hidden layers with 32, 16, and 8 neurons, respectively to best approximate $\hat{\varepsilon}_{AHBS,j,i}$ by minimizing

$$\frac{1}{n_i}\sum_{j=1}^{n_i}\left[g(\tau_j, m_j) - \hat{\varepsilon}_{AHBS,j,i}\right]^2,$$

In an h-step ahead forecasting, the IVS predicted by the AHBS-DNN model is given by the sum of the projected value of the AHBS model and its neural network correction: $\widehat{IV}_{i+h}(\tau_j, m_j) + \hat{g}(\tau_j, m_j), j = 1, ..., n_{i+h}$.

Ad-hoc Black–Scholes with Vector Autoregressive (AHBS-VAR). Goncalves and Guidolin (2006) propose a vector autoregressive approach to model the dynamics of the parameters of the Ad-hoc Black–Scholes (AHBS) model of Dumas, Fleming, and Whaley (1998) and predict their values in the future. A VAR model is fitted to capture the dynamics of $\boldsymbol{\alpha}_i = (\alpha_{0,i}, \alpha_{1,i}, \alpha_{2,i}, \alpha_{3,i}, \alpha_{4,i}, \hat{\alpha}_{5,i})^T$:

$$\boldsymbol{\alpha}_{i+h} = \boldsymbol{\mu} + \boldsymbol{\Phi}_1\boldsymbol{\alpha}_i + \frac{1}{5}\boldsymbol{\Phi}_5\sum_{j=i-4}^{i}\boldsymbol{\alpha}_j + \frac{1}{22}\boldsymbol{\Phi}_{22}\sum_{j=i-21}^{i}\boldsymbol{\alpha}_j + \boldsymbol{\varepsilon}_{i+h}, \tag{A.35}$$

where $\boldsymbol{\varepsilon}_{i+h} \overset{\text{i.i.d}}{\sim} N(0, \Xi)$. After predicting $\hat{\boldsymbol{\alpha}}_{i+h}$ with equation (A.35), the implied volatility predicted by the AHBS model is attained by using $\hat{\boldsymbol{\alpha}}_{i+h}$ and equation (A.34).

A.8.4 Autoencoder with long short- term memory model

We adapt the modeling framework of Zhang, Li, and Zhang (2023), using a two-step framework for forecasting IVS. First, we use basis spline functions to interpolate IVS for each day. The interpolation is conducted with orthogonal splines of degree 3, on the grid of moneyness from -2 to 2 by a step of 0.2, and time-to-maturity between 5 and 252 days by a step of 11 days. This results in a total of 483 IV values for each day. To reduce the high spatial dimension and extract meaningful hidden features for each IVS, we employ autoencoder (AE), a generalization of principal component analysis (PCA), as shown in Gu, Kelly, and Xiu (2021).

Denote $X_i \in \mathbb{R}^{483}$ as the input vector (i.e., the vectorized IV grid values of day i). The AE consists of two primary components: the encoder and the decoder. The encoder maps the high-dimensional input data to a lower-dimensional latent space, while the decoder reconstructs the original data from this compressed representation. The encoder is a fully connected neural network of three layers, each with 512 neurons and ReLU activation function, reducing the input to a latent representation $Z \in \mathbb{R}^d$ where $d = 8$. The decoder attempts to reconstruct the original input X from the compressed latent representation Z_i. It mirrors the encoder's structure, increasing the dimensionality back to the original input size. The encoder and decoder can be written as

$$\begin{aligned}
\text{Encoder :} Z_i &= f_{\text{enc}}(X_i) = ReLU(W_3 ReLU(W_2 ReLU(W_1 X_i + b_1) + b_2) + b_3) \\
\text{Decoder :} \hat{X}_i &= f_{\text{dec}}(Z_i) = ReLU(W_6 ReLU(W_5 ReLU(W_4 Z_i + b_4) + b_5) + b_6),
\end{aligned}$$

where W_j and $b_j, j = 1,2,3$ are the weights and biases for the respective hidden layers in the encoder; W_j and $b_j, j = 4,5,6$ are the weights and biases for the respective hidden layers in the decoder; and $\hat{X}_i$ is the reconstructed output that has the same dimension as X_i. The AE is trained by minimizing the squared error between the input X_i and the reconstructed output $\hat{X}_i$

$$\mathscr{L} = \sum_{i=1}^{n} \|X_i - \hat{X}_i\|^2.$$

In h-step ahead forecasting, to predict Z_{i+h}, i.e., the latent features of day $i+h$, the HAR framework and the long short-term (LSTM) model of Hochreiter and Schmidhuber (1997) are utilized. The input used to forecast Z_{i+h} consists of $Z_i^1 = \frac{1}{22}\sum_{k=i-21}^{i} Z_k$, $Z_i^2 = \frac{1}{5}\sum_{k=i-4}^{i} Z_k$ and $Z_i^3 = Z_i$. Denote h_j a hidden state representing a summary information from $\{Z_i^1, ..., Z_i^j\}$, and

$h_0 = 0$. For each $j = 1,2,3$, recursively compute

$$\begin{aligned}
r_j &= \sigma_g(W_r Z_i^j + U_r h_{j-1} + b_r),\\
n_j &= \sigma_g((W_i Z_i^j + U_i h_{j-1} + b_i),\\
o_j &= \sigma_g((W_o Z_i^j + U_o h_{j-1} + b_o),\\
g_j &= \sigma_h((W_g Z_i^j + U_g h_{j-1} + b_g),\\
c_j &= r_j \odot c_{j-1} + n_j \odot g_j,\\
h_j &= o_j \odot \sigma_h(c_j),\\
y_j &= \sigma_h(W_y h_j + b_y),
\end{aligned}$$

where W, U, b are parameters, and σ_g, σ_h are the sigmoid activation function $\frac{1}{1+\exp(-x)}$ and the tanh activation function $\frac{1-\exp(-2x)}{1+\exp(-2x)}$, respectively. At each j value, n_j, r_j, and o_j represent the input, forget, and output gates, respectively. The final output of the LSTM model is the predicted value for Z_{i+h}

$$\hat{Z}_{i+h} = W_{out} y_3 + b_{out}.$$

The LSTM model has 100 hidden states, i.e., $r_j, n_j, o_j \in (0,1)^{100}$, and $h_j \in (-1,1)^{100}$. It is trained over 200 epochs with a learning rate of 0.01 and batch size of 128, and a squared loss function

$$\mathscr{L} = \sum_{i=1}^{n} \|Z_{i+h} - \text{LSTM}(Z_i^1, Z_i^2, Z_i^3)\|^2.$$

After obtaining the predicted $\hat{Z}_{i+h}$ from the LSTM model, we pass $\hat{Z}_{i+h}$ into the decoder of the trained AE model and get the predicted interpolated IV values $\hat{\hat{X}}_i$ on the specified grid. Using $\hat{\hat{X}}_i$ and orthogonal cubic splines, we can then obtain the predicted observed IV values on day $i+h$.

A.9 Additional results on statistical performance

On top of RMSE and OoR^2 reported in the main text, we assess prediction accuracy from two more angles, error magnitude, and directional changes, both derived from observed test data. Mean absolute percentage error (MAPE) captures prediction accuracy and error magnitude, while mean correct prediction of direction of change (MCPDC) evaluates the ability of the models to anticipate the direction of price movements. These two metrics contribute another unique insight into the performances of the models.

$$\begin{aligned}\text{MAPE}_h &= \frac{1}{\sum_{i=i_0}^{n-h} n_i} \sum_{i=i_0}^{n-h} \sum_{j=1}^{n_i} \left| \frac{Y_i(\tau_j, m_j) - \hat{Y}_i(\tau_j, m_j)}{Y_i(\tau_j, m_j)} \right|, \\ \text{MCPDC}_h &= \frac{1}{\sum_{i=i_0}^{n-h} n_i} \sum_{i=i_0}^{n-h} \sum_{j=1}^{n_i} \times \mathbb{1}_{(IV_{i+h}(\tau_j, m_j) - IV_i(\tau_j, m_j))(\widehat{IV}_{i+h}(\tau_j, m_j) - IV_i(\tau_j, m_j)) > 0}.\end{aligned}$$

Here, n_i denotes the number of observed options on day i, and $i_0 = 2523$ and $n = 3273$ mark the start and end of the testing period. The MCPDC$_h$ is computed solely for options traded on both day i and day $i+h$. For consistency, we use a different notation in the MCPDC formula, where IV_i and $\widehat{IV}_i$ denote observed and forecasted IV values of day i, respectively.

	RMSE					MAPE					OoR^2					MCPDC				
	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$
Overall (from Jan 9, 2019 to Dec 31, 2021)																				
DNN-RW	3.62	5.61	7.56	9.15	10.40	8.73	12.86	16.44	18.86	20.83	88.45	72.62	50.43	27.68	6.67	41.62	42.13	42.41	44.50	47.01
CW-RW	3.57	5.45	7.23	8.87	9.91	8.46	12.49	16.05	18.36	20.23	88.74	74.18	54.50	32.01	15.21	41.80	43.40	45.35	49.11	51.85
CW-DNN	**3.51**	5.47	7.29	8.97	10.03	7.80	12.01	15.69	18.09	20.00	**89.11**	74.00	53.69	30.47	13.27	41.41	44.40	46.21	50.21	52.68
AHBS-RW	3.63	5.68	7.65	9.26	10.54	8.24	12.61	16.31	18.78	20.82	88.39	71.90	49.17	25.87	4.16	41.21	43.19	44.87	47.96	50.40
AHBS-DNN	3.86	5.83	7.75	9.34	10.62	8.79	12.97	16.55	18.98	21.00	86.91	70.39	47.81	24.60	2.76	42.20	43.14	44.84	47.66	50.05
AHBS-VAR	**3.51**	5.37	7.30	8.68	9.60	7.95	12.10	15.10	17.22	18.60	**89.11**	74.93	53.72	34.93	20.49	44.74	45.30	49.32	52.38	54.68
AE-LSTM	3.65	4.80	6.20	7.75	8.97	8.21	11.70	13.69	14.60	16.15	88.27	79.98	66.51	48.08	30.44	47.62	51.01	54.01	58.76	59.46
fRW	3.71	5.75	7.69	9.30	10.61	7.32	11.97	15.87	18.42	20.51	87.92	71.23	48.57	25.23	2.84	39.66	44.59	46.26	49.69	51.47
fLinK	3.66	5.41	7.37	8.52	9.47	**7.01**	11.51	14.64	16.58	17.78	88.20	74.51	52.70	37.16	22.42	44.02	46.71	51.06	54.59	56.81
fGauK	6.95	6.92	6.95	7.13	7.75	8.45	11.02	12.45	13.02	13.19	57.84	58.11	58.02	56.05	48.27	45.34	55.40	62.12	65.88	68.98
fLapK	6.96	7.01	6.93	7.12	7.04	8.69	10.73	11.51	12.04	12.01	57.77	57.19	58.42	56.22	57.35	**46.36**	55.51	**64.48**	**68.60**	**72.49**
fNTK	3.80	**4.73**	**5.45**	**5.77**	**5.74**	7.39	**9.69**	**10.39**	**11.07**	**10.89**	87.31	**80.39**	**74.02**	**71.26**	**71.62**	44.74	**56.76**	64.46	68.34	72.42
From Jan 9, 2019 to Dec 31, 2019																				
DNN-RW	1.59	2.36	2.82	3.10	3.40	7.64	11.43	14.13	15.56	17.69	85.11	67.05	52.92	43.13	32.33	40.95	40.44	42.52	47.14	49.29
CW-RW	1.53	2.23	2.65	2.91	3.19	7.57	11.24	13.90	15.34	17.44	86.13	70.52	58.33	50.05	40.40	40.90	40.98	44.85	50.52	53.52
CW-DNN	1.48	2.23	2.66	2.92	3.21	6.92	10.80	13.56	14.94	17.09	87.02	70.66	58.09	49.56	39.52	40.62	41.27	45.55	51.52	54.25
AHBS-RW	1.59	2.42	2.90	3.19	3.50	7.34	11.32	14.22	15.72	17.94	85.15	65.51	50.36	39.82	28.10	40.57	41.14	44.42	50.32	52.59
AHBS-DNN	1.67	2.47	2.94	3.23	3.54	7.82	11.64	14.40	15.88	18.06	83.56	64.04	49.00	38.45	26.61	41.91	40.90	44.00	49.83	51.86
AHBS-VAR	1.54	2.25	2.65	2.87	2.95	7.16	10.77	13.24	14.69	15.34	86.06	70.20	58.44	51.20	48.82	**44.49**	45.92	49.80	54.16	57.07
AE-LSTM	1.62	2.30	2.55	2.90	2.89	7.47	10.88	12.36	14.61	15.06	84.48	68.53	61.54	50.09	50.97	48.13	51.67	54.59	55.77	57.53
fRW	1.53	2.40	2.91	3.21	3.53	6.48	10.79	13.90	15.46	17.74	86.20	65.90	50.04	39.24	27.12	38.98	41.98	45.46	51.84	53.52
fLinK	**1.41**	2.18	2.48	2.75	2.87	**6.17**	9.87	11.51	12.59	13.25	**88.19**	72.02	63.56	55.33	51.82	44.31	46.63	51.84	53.82	53.93
fGauK	1.46	2.02	2.26	2.35	2.26	6.38	9.47	11.09	12.13	11.39	87.36	75.92	69.66	67.29	69.66	43.00	51.66	59.09	63.93	68.54
fLapK	1.46	**1.88**	2.00	**2.01**	**1.93**	6.49	**8.89**	9.89	**10.27**	**9.69**	87.39	**79.03**	76.33	**76.10**	**78.09**	44.46	**55.55**	62.29	68.05	70.97
fNTK	1.57	2.02	**1.99**	2.07	2.08	6.75	9.06	**9.49**	**10.27**	10.07	85.51	75.98	**76.63**	74.45	74.32	44.02	53.75	**64.10**	**68.69**	**71.08**
From Jan 1, 2020 to Dec 31, 2020																				
DNN-RW	5.27	8.24	11.36	13.91	15.88	9.39	14.68	20.45	24.39	27.35	83.05	58.74	19.58	-22.12	-62.13	42.63	43.82	41.53	41.74	41.98
CW-RW	5.29	8.11	10.95	13.57	15.19	9.35	14.52	20.28	24.00	26.71	82.92	59.91	24.89	-16.35	-48.49	42.92	45.15	44.40	46.96	47.14
CW-DNN	5.22	8.15	11.06	13.74	15.38	8.72	14.06	19.99	23.87	26.71	83.31	59.51	23.37	-19.17	-52.08	42.00	46.05	44.49	47.38	47.30
AHBS-RW	5.34	8.36	11.51	14.08	16.08	9.36	14.76	20.55	24.49	27.54	82.61	57.48	17.59	-25.09	-66.39	43.11	44.40	43.11	44.74	44.30
AHBS-DNN	5.70	8.60	11.66	14.20	16.20	10.12	15.26	20.87	24.75	27.77	80.27	55.01	15.39	-27.17	-68.72	44.06	44.15	43.40	44.41	44.06
AHBS-VAR	**5.17**	7.92	11.03	13.22	14.67	8.93	13.93	18.08	20.80	22.71	**83.64**	61.85	24.33	-10.26	-38.14	46.34	45.37	49.08	51.23	51.38
AE-LSTM	5.39	6.97	9.23	11.81	13.77	9.42	12.78	15.34	17.42	20.71	82.19	70.40	46.68	11.83	-22.81	49.45	52.02	58.00	61.78	58.80
fRW	5.52	8.48	11.58	14.15	16.20	8.55	14.13	20.13	24.22	27.34	81.42	56.22	16.47	-26.39	-69.00	41.74	46.77	44.97	46.86	45.79
fLinK	5.50	8.03	11.17	12.98	14.51	**8.16**	13.41	17.96	20.89	22.95	81.43	60.71	21.86	-7.09	-36.35	45.79	47.71	51.94	55.14	55.07
fGauK	11.01	10.77	10.71	10.90	11.92	11.93	14.34	16.14	16.15	17.21	26.97	28.44	28.29	24.79	8.91	48.20	58.74	66.63	72.05	71.89
fLapK	11.02	10.97	10.76	11.00	10.86	12.44	14.61	15.67	15.88	16.43	26.88	26.43	28.29	23.71	24.25	**49.63**	56.81	67.02	71.52	73.15
fNTK	5.70	**7.07**	**8.29**	**8.71**	**8.67**	8.69	**11.42**	**12.76**	**13.05**	**13.19**	80.17	**69.00**	**56.61**	**52.14**	**51.77**	46.19	**60.03**	**67.11**	**72.89**	**74.66**
From Jan 1, 2021 to Dec 31, 2021																				
DNN-RW	2.20	3.18	3.61	3.81	3.93	8.74	11.91	13.75	15.14	15.83	87.14	73.22	65.44	61.29	57.10	41.01	41.50	43.24	45.73	50.75
CW-RW	1.97	2.81	3.22	3.40	3.49	8.12	11.21	13.03	14.35	15.05	89.69	78.95	72.39	69.14	66.14	41.22	43.15	46.61	50.49	55.60
CW-DNN	1.89	2.80	3.21	3.40	3.49	7.42	10.68	12.60	14.00	14.65	90.52	79.26	72.53	69.18	66.18	41.30	44.69	48.34	52.32	57.16
AHBS-RW	2.08	3.17	3.64	3.85	3.97	7.68	11.23	13.23	14.70	15.43	88.51	73.34	64.79	60.55	56.23	39.69	43.23	46.90	49.80	55.21
AHBS-DNN	2.14	3.21	3.68	3.89	4.01	8.04	11.48	13.44	14.87	15.60	87.74	72.63	64.05	59.81	55.47	40.51	43.49	46.80	49.64	54.98
AHBS-VAR	2.00	2.96	3.38	3.58	3.69	7.46	11.05	13.18	15.04	16.28	89.39	76.70	69.75	65.88	62.21	43.28	44.83	49.21	52.47	56.55
AE-LSTM	1.99	2.78	3.15	2.98	3.09	7.47	11.12	12.82	11.62	11.98	89.45	79.47	73.64	76.40	73.47	45.48	49.58	49.74	57.59	61.24
fRW	1.98	3.15	3.64	3.84	3.96	6.61	10.51	12.71	14.20	14.94	89.64	73.74	64.74	60.84	56.57	38.01	44.06	48.03	51.26	55.95
fLinK	**1.85**	2.92	3.43	3.55	3.50	6.38	10.60	13.22	14.62	15.19	**90.94**	77.45	68.75	66.41	66.06	42.08	45.76	49.65	54.40	60.16
fGauK	1.86	2.43	2.56	2.82	2.70	**6.25**	8.62	9.50	10.28	10.03	90.85	84.32	82.59	78.91	79.86	43.92	**54.46**	59.67	61.12	66.42
fLapK	**1.85**	**2.27**	**2.32**	**2.49**	**2.31**	6.28	**7.92**	**8.23**	**9.11**	**8.75**	**90.94**	**86.35**	**85.69**	**83.51**	**85.24**	**44.27**	54.22	**63.38**	**66.12**	**72.73**
fNTK	1.90	2.38	2.35	2.61	2.43	6.49	8.34	8.50	9.49	8.93	90.44	84.95	85.34	81.84	83.63	43.75	55.38	62.17	63.80	71.01

Table A.3: Prediction accuracy RMSE, MAPE, OoR^2 and MCPDC of all models over the whole test period (Jan 09, 2019 to Dec 31, 2021) and in each of three years 2019 (before Covid), 2020 (Covid year) and 2021 (recovery year). In this table, fNTKℓ refers to the fNTK model with ℓ hidden layers, and fNTK3 is the fNTK in the main text. Bold numbers indicate the best-performing model (or models) in a given column.

A.9.1 Robustness of statistical performance

Varying degrees of orthogonal splines. Besides splines basis, there are many other choices of (orthogonal) basis functions, such as Fourier (Chen and Li, 2017), or wavelets (Morris et al., 2003). However, Fourier basis functions are more suitable for periodic data, and wavelets are useful for wiggly curves. Given the non-periodic, smooth nature of IVS, we only consider splines bases in our empirical study. For simplicity, the degrees of splines in the time-to-maturity and moneyness directions are assumed to be the same. We tested different degrees 2, 3, 4, 5, and 6, which correspond to 9, 16, 25, 36, and 49 total number of basis functions, respectively. Table A.4 shows that while we often benefit from using a degree higher than 2, degrees that are larger than 3 may only provide marginal improvements and may even worsen the results. For example, in the 20-step ahead horizon, using degree 3 instead of 2 reduces MAPE from 11.44% to 10.89%, and degree 4 marginally improves it to 10.83%. But the MAPE worsens when we use degree 5 (12.24%) or degree 6 (59.07%).

Degree	# bases	RMSE					MAPE				
		$h=1$	$h=5$	$h=10$	$h=15$	$h=20$	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$
2	9	4.03	4.95	5.32	5.87	**5.61**	8.64	10.76	11.06	11.67	11.44
3	16	**3.80**	**4.73**	5.45	**5.77**	5.74	7.39	**9.69**	10.39	**11.07**	10.89
4	25	3.83	5.21	**5.09**	6.04	5.73	**7.29**	9.84	**10.32**	11.08	**10.83**
5	36	4.60	5.88	5.63	6.12	5.77	8.32	11.04	11.53	12.10	12.24
6	49	26.41	30.91	24.81	21.93	30.35	46.20	59.40	48.71	48.94	59.07

Degree	# bases	OoR^2					MCPDC				
		$h=1$	$h=5$	$h=10$	$h=15$	$h=20$	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$
2	9	85.71	78.58	75.22	70.12	**72.87**	45.90	54.42	62.09	66.01	69.82
3	16	**87.31**	**80.39**	74.02	**71.26**	71.62	44.74	**56.76**	64.46	68.34	72.42
4	25	86.98	76.21	**77.54**	68.50	71.70	45.02	56.70	**65.28**	**68.73**	**73.19**
5	36	81.50	69.93	72.50	67.62	71.30	45.05	54.52	63.42	67.29	69.70
6	49	-510.23	-742.15	-440.81	-320.05	-872.32	**48.25**	50.36	53.45	55.48	56.92

Table A.4: Prediction accuracy of fNTK (with three hidden layers) using different degrees of orthogonal splines in terms of RMSE, MAPE, OoR^2, and MCPDC. The degree column indicates the degree of the splines, and the number of bases column indicates the total number of basis functions corresponding to each degree. The prediction period is from Jan 09, 2019 to Dec 31, 2021. Bold numbers indicate the best-performing model (or models) in a given column.

Effects of HAR lags. In the main text, to forecast the basis coefficients $\boldsymbol{y}_i$ of $Y_i = IV_{i+h}$, we incorporate basis coefficients of different lags of implied volatility surfaces, using the Heterogeneous

Autoregressive (HAR) framework (Corsi, 2009). Let $\boldsymbol{x}_i^{(d)}$, $\boldsymbol{x}_i^{(w)}$, and $\boldsymbol{x}_i^{(m)}$ be the basis coefficients of $X_i^{(d)} = IV_i$, $X_i^{(w)} = \frac{1}{5}\sum_{k=i-4}^{i} IV_k$ and $X_i^{(m)} = \frac{1}{22}\sum_{k=i-21}^{i} IV_k$, then we have the predictor vector $\boldsymbol{x}_i = (\boldsymbol{x}_i^{(d)}, \boldsymbol{x}_i^{(w)}, \boldsymbol{x}_i^{(m)})^T$. The empirical advantage of using the HAR framework is shown in Table A.5, where we compare fNTK using the HAR setup versus using the 22 lags, i.e., basis coefficients of $IV_{i-21}, IV_{i-20}, \ldots, IV_i$ are concatenated into $\boldsymbol{x}_i$, without taking average. It is evident that in most forecasting horizons, using the HAR framework tends to give better results than the 22 lags separately as individual predictors. This may be induced by averaging in the HAR framework that gives a good balance between extracting short/medium/long features while smoothing out the noise present in each of the 22 lags.

	RMSE					MAPE				
	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$
fNTK with HAR lags	3.80	**4.73**	**5.45**	**5.77**	**5.74**	**7.39**	**9.69**	**10.39**	**11.07**	**10.89**
fNTK with 22 lags	**3.52**	5.12	6.15	6.07	6.10	7.82	10.36	11.43	11.56	11.80
	OoR^2					MCPDC				
	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$
fNTK with HAR lags	87.31	**80.39**	**74.02**	**71.26**	**71.62**	44.74	**56.76**	**64.46**	**68.34**	**72.42**
fNTK with 22 lags	**89.13**	77.21	67.26	68.20	67.88	**46.45**	56.04	62.35	68.20	71.53

Table A.5: Prediction accuracy of fNTK with HAR lags and with 22 lags in terms of RMSE, MAPE, OoR^2, and MCPDC. The prediction period is from Jan 09, 2019 to Dec 31, 2021. Bold numbers indicate the best-performing model (or models) in a given column.

Varying number of hidden layers. The outcomes documented in Table A.6 underscore an intriguing observation: while utilizing a Neural Tangent Kernel (NTK) framework with three hidden layers contributes to improved prediction accuracy, the benefits of incorporating a higher number of hidden layers are not uniformly positive. For example, when focusing on the forecasting horizon of $h = 15$, the Root Mean Squared Error (RMSE) diminishes from 7.23% to 5.77% whereas employing five layers slightly exacerbates RMSE to 5.82%.

Varying number of training samples. We investigate the models' performances when using different numbers of training samples, 1000 and 2000. Table A.7 shows that the functional models are robust to the training size, with fNTK still leading the best performance across the longer horizons when using a training size of either 1000 or 2000.

Options in different moneyness and tim-to-maturity categories. We categorize options into

	RMSE					MAPE				
	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$
fNTK1	**3.52**	5.06	6.30	7.23	7.91	**7.08**	10.59	12.88	14.19	14.76
fNTK3	3.80	4.73	**5.45**	**5.77**	5.74	7.39	9.69	10.39	11.07	10.89
fNTK5	3.66	**4.63**	5.49	5.82	**5.51**	7.63	**9.50**	**9.91**	**10.49**	**10.22**
	OoR^2					MCPDC				
	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$
fNTK1	**89.16**	77.58	65.23	54.72	45.92	44.73	51.23	56.02	60.50	62.98
fNTK3	87.31	80.39	**74.02**	**71.26**	71.62	44.74	56.76	64.46	68.34	72.42
fNTK5	88.26	**81.22**	73.74	70.71	**73.83**	**45.19**	**57.78**	**66.87**	**69.76**	**74.62**

Table A.6: Prediction accuracy of fNTK using different numbers of hidden layers (1, 3, and 5) in terms of RMSE, MAPE, OoR^2, and MCPDC. The prediction period is from Jan 09, 2019 to Dec 31, 2021. Bold numbers indicate the best-performing model (or models) in a given column.

distinct groups based on their moneyness (m) and time-to-maturity (τ), subsequently examining the statistical performance of the models within each group. With respect to moneyness, the options are stratified into four groups: $[-2,-0.5]$, $(-0.5,0]$, $(0,0.5]$, and $(0.5,2]$, while maturity is divided into four intervals based on days to maturity: $[5,60]$, $(60,120]$, $(120,180]$, and $(180,252]$ days. The performance of the models across moneyness and time-to-maturity groups, as measured by the Root Mean Square Error (RMSE), is illustrated in Figure A.3.[4]

The analysis reveals that, on the whole, prediction errors, as quantified by RMSE, tend to be higher for options with larger moneyness (m) and shorter maturity (τ). Remarkably, the consistent pattern across all moneyness and time-to-maturity groups, as well as prediction periods, is the superior performance of nonlinear functional models, notably exemplified by fNTK, particularly in longer forecasting horizons. Notably, across the entirety of the analysis, all models exhibit diminished forecasting accuracy in the year 2020 compared to 2019 and 2021.

These findings collectively underscore the robustness of nonlinear models across various market conditions and reveal nuanced insights into the interplay between model complexity and performance across distinct prediction horizons.

[4] We tried the put and call options separately and the same conclusions hold. Hence, it does not seem to be a matter of liquidity. Additional results for put and calls separately are available upon request.

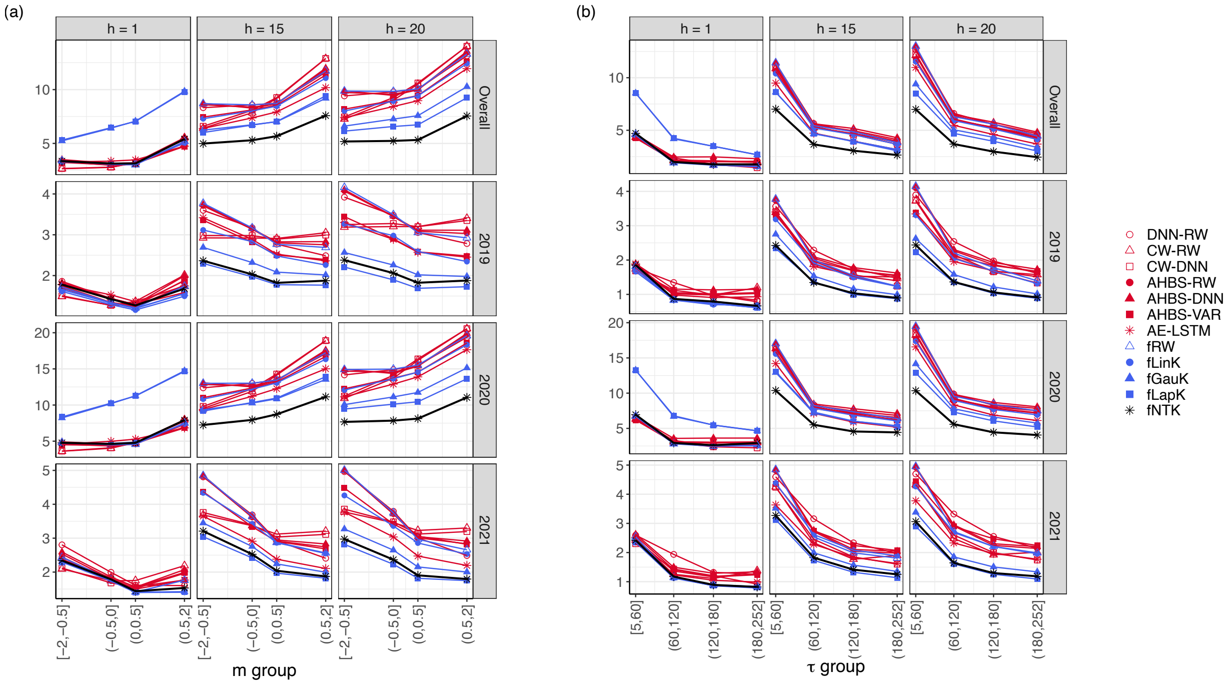


Figure A.3: Prediction accuracy in terms of RMSE of all the models across forecasting horizons $h = 1, 15$ and 20, in (a) four different moneyness m values: $[-2,-0.5], (-0.5,0], (0,0.5]$, and $(0.5,2]$ and (b) time-to-maturity τ values: $[5,60], (60,120], (120,180]$, and $(180,252]$ days. The performance period is split into overall (from Jan 09, 2019, to Dec 31, 2021), year 2019, year 2020, and year 2021.

Panel A: Training size of 1000 days

	RMSE					MAPE				
	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$
DNN-RW	3.62	5.61	7.56	9.15	10.40	8.73	12.86	16.44	18.86	20.83
CW-RW	3.57	5.45	7.23	8.87	9.91	8.46	12.49	16.05	18.36	20.23
CW-DNN	**3.51**	5.47	7.29	8.97	10.03	**7.80**	12.01	15.69	18.09	20.00
AHBS-RW	3.63	5.68	7.65	9.26	10.54	8.24	12.61	16.31	18.78	20.82
AHBS-DNN	3.86	5.83	7.75	9.34	10.62	8.79	12.97	16.55	18.98	21.00
AHBS-VAR	3.69	5.46	7.57	8.80	9.67	8.20	12.49	15.84	17.80	18.64
AE-LSTM	3.99	4.98	5.75	6.43	6.61	8.09	11.35	12.53	13.74	13.69
fRW	3.71	5.75	7.69	9.30	10.61	7.32	11.97	15.87	18.42	20.51
fLinK	4.09	5.74	7.48	8.32	9.27	7.46	12.30	16.42	18.17	19.43
fGauK	8.41	9.05	8.13	7.71	7.29	9.77	12.40	12.92	13.21	13.11
fLapK	8.53	8.67	8.35	8.08	7.48	10.26	11.99	12.61	12.94	12.86
fNTK	4.43	**5.10**	**5.43**	**5.75**	**5.58**	8.05	**9.60**	**9.89**	**10.27**	**10.06**

	OoR^2					MCPDC				
	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$
DNN-RW	88.45	72.62	50.43	27.68	6.67	41.62	42.13	42.41	44.50	47.01
CW-RW	88.74	74.18	54.50	32.01	15.21	41.80	43.40	45.35	49.11	51.85
CW-DNN	**89.11**	74.00	53.69	30.47	13.27	41.41	44.40	46.21	50.21	52.68
AHBS-RW	88.39	71.90	49.17	25.87	4.16	41.21	43.19	44.87	47.96	50.40
AHBS-DNN	86.91	70.39	47.81	24.60	2.76	42.20	43.14	44.84	47.66	50.05
AHBS-VAR	88.00	74.11	50.22	33.03	19.35	45.11	45.14	48.02	49.83	53.64
AE-LSTM	86.08	**78.30**	71.16	64.19	62.23	46.11	50.73	58.30	60.17	63.81
fRW	87.92	71.23	48.57	25.23	2.84	39.66	44.59	46.26	49.69	51.47
fLinK	85.40	71.31	51.15	40.13	25.96	44.51	48.11	52.12	56.54	60.30
fGauK	38.15	28.60	42.58	48.10	54.11	45.56	54.63	62.29	66.54	69.79
fLapK	36.42	34.39	39.44	43.59	51.73	**47.07**	53.96	63.45	68.05	72.14
fNTK	82.87	77.11	**74.33**	**71.48**	**72.99**	44.18	**59.10**	**66.89**	**72.33**	**76.65**

Panel B: Training size of 2000 days

	RMSE					MAPE				
	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$
DNN-RW	3.62	5.61	7.56	9.15	10.40	8.73	12.86	16.44	18.86	20.83
CW-RW	3.57	5.45	7.23	8.87	9.91	8.46	12.49	16.05	18.36	20.23
CW-DNN	**3.51**	5.47	7.29	8.97	10.03	7.80	12.01	15.69	18.09	20.00
AHBS-RW	3.63	5.68	7.65	9.26	10.54	8.24	12.61	16.31	18.78	20.82
AHBS-DNN	3.86	5.83	7.75	9.34	10.62	8.79	12.97	16.55	18.98	21.00
AHBS-VAR	3.58	5.35	7.38	8.69	9.56	8.01	12.05	14.83	16.47	17.33
AE-LSTM	4.04	4.78	5.77	6.76	7.77	8.07	11.40	12.96	13.77	13.96
fRW	3.71	5.75	7.69	9.30	10.61	7.32	11.97	15.87	18.42	20.51
fLinK	3.91	5.56	7.49	8.39	9.14	**7.10**	11.61	14.91	16.71	17.90
fGauK	8.06	8.15	7.75	7.44	7.46	9.39	11.59	12.54	12.97	12.97
fLapK	8.20	8.29	7.99	7.82	7.40	9.91	11.66	12.28	12.74	12.62
fNTK	4.26	**4.99**	**5.33**	**5.70**	**5.39**	7.55	**9.57**	**9.92**	**10.52**	**10.18**

	OoR^2					MCPDC				
	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$
DNN-RW	88.45	72.62	50.43	27.68	6.67	41.62	42.13	42.41	44.50	47.01
CW-RW	88.74	74.18	54.50	32.01	15.21	41.80	43.40	45.35	49.11	51.85
CW-DNN	**89.11**	74.00	53.69	30.47	13.27	41.41	44.40	46.21	50.21	52.68
AHBS-RW	88.39	71.90	49.17	25.87	4.16	41.21	43.19	44.87	47.96	50.40
AHBS-DNN	86.91	70.39	47.81	24.60	2.76	42.20	43.14	44.84	47.66	50.05
AHBS-VAR	88.68	75.11	52.72	34.69	21.24	44.68	44.87	48.64	51.46	55.01
AE-LSTM	85.67	**80.04**	70.98	60.54	47.97	46.55	49.75	57.99	59.79	64.32
fRW	87.92	71.23	48.57	25.23	2.84	39.66	44.59	46.26	49.69	51.47
fLinK	86.60	73.08	51.24	39.13	27.96	43.82	47.57	52.24	56.08	58.69
fGauK	43.26	41.99	47.75	51.96	52.07	45.64	56.07	64.22	68.17	71.05
fLapK	41.34	40.19	44.72	47.19	52.78	**46.84**	54.81	64.16	68.56	72.38
fNTK	84.16	78.02	**75.22**	**71.96**	**74.99**	44.90	**58.03**	**66.60**	**70.91**	**75.00**

Table A.7: Prediction accuracy of all models in terms of RMSE, MAPE, OoR^2, and MCPDC. The models are trained with two different training sizes: 1000 days in Panel A and 2000 days in Panel B. The prediction period is from Jan 09, 2019 to Dec 31, 2021. Bold numbers indicate the best-performing model (or models) in a given column.

A.9.2 Diebold-Mariano tests

We use the Diebold-Mariano (DM) test to determine whether the two prediction performances are significantly different, see Diebold and Mariano (2002). Let $e_{h,1i}$ and $e_{h,2i}$ denote the $h-$day forecasting error of model 1, and model 2 at time $i \in [i_0, n-h]$. Since we have multiple observations and forecasted values on each test day i, the mean squared error is defined as $e_{h,i} = \sqrt{\frac{1}{n_i}\sum_{j=1}^{n_i}(Y_{i+h}(\tau_j, m_j) - \hat{Y}_{i+h}(\tau_j, m_j))^2}$. The differential loss is computed by $d_{h,i} = e_{h,1i} - e_{h,2i}$, and the DM test statistic is computed as follows

$$\mathrm{DM}_h = \frac{\frac{1}{n-h-i_0+1}\sum_{i=i_0}^{n-h} d_{h,i}}{\sqrt{(\hat{\sigma}_0 + 2\sum_{k=1}^{h}\hat{\sigma}_k)/(n-h-i_0-22+1)}},$$

where n_i denotes the number of observed options on day i, $i_0 = 2523$ and $n = 3273$ mark the start and end of the testing period, $\hat{\sigma}_0$ is the sample standard deviation, and $\hat{\sigma}_k$ is the autocovariance at lag $k \geq 1$ of the series d_i^h. The null hypothesis $H_0 : \mathrm{DM}_h = 0$ is that there is no significant difference in the accuracy of the two models, and the one-sided alternative $H_1 : \mathrm{DM}_h > 0$ is that method 2 is more accurate than method 1. In Table 1, method 1 refers to our proposed model fNTK and method 2 refers to an alternative model. The reported significance levels of $10\%, 5\%$, and 1% are based on the minimum of the p-values of the DM tests for put and call options separately.

A.9.3 Forecasting with interpolated implied volatility grid values

Given a set of random, non-equidistant, discretely observed values of implied volatility surfaces, there are two different approaches for tackling the unbalanced design and evaluating the curves. We can follow the approach used in this paper and utilize a set of basis functions to fit the surfaces and obtain corresponding basis coefficients. Alternatively, we can evaluate the surfaces on the same grid of moneyness and time-to-maturity values using a set of basis functions and attain the IV values at the grid locations. For example, given a set of basis functions, Ramsay and Silverman (2005) presents two methods to perform functional principal component analysis on observed curves: one approach proposes to work with (multivariate) PCA on the discretized (interpolated) values, another approach is to work with the basis coefficients. Their empirical examples show that there is little discrepancy between the results of the two methods. While these two approaches rely on the same amount of information about the surfaces, evaluating the functions on a fixed grid requires an extra step: after interpolation, we need to evaluate the surface at the grid locations using the inner product between the basis functions and their coefficients. This approach also requires a higher dimension to summarize the curves (i.e., equal to the number of interpolated points) relative to directly storing the basis coefficients.

For a robustness check, we look at the prediction accuracy when using the interpolated IV values on a fixed grid with three neural network forecasting models: the NTK parameterized fully connected neural network, the standard fully connected deep neural network (DNN), and the long-short term memory model (LSTM). The DNN has the same architecture as the NN with NTK parameterization: three hidden layers with 500 neurons in each layer; the LSTM has 100 hidden units in its hidden layer. The interpolation is conducted with orthogonal splines of degree 3, on the grid of moneyness from -2 to 2 by a step of 0.2, and time-to-maturity between 5 and 252 days by a step of 11 days. This results in a total of 483 IV values for each day. Table A.8 shows the prediction accuracy the three models across the forecasting horizons $h = 1, 5, 10, 15$ and 20[5]. The three models have highly similar performance, and highly comparable with the fNTK results in our main text. This further validates the consistency between using interpolated values and using basis coefficients.

A.9.4 Forecasting with dimension reduction of interpolated implied volatility grid values

We adapt the modeling framework of Zhang, Li, and Zhang (2023), using a two-step framework for forecasting IVS. First, they use the AHBS model to obtain interpolated IV values on a grid of moneyness and time-to-maturity, similar to what we have in Section A.9.3, except that we use a more flexible semi-parametric approach. Two dimension-reduction approaches are then employed to reduce the dimension of the interpolated values: principal component analysis (PCA) and au-

[5] For fair comparison, all models are evaluated on observed IV values.

	RMSE					MAPE				
	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$
NTK	**3.81**	4.68	5.50	**5.60**	**5.21**	**7.62**	**9.37**	**9.87**	**10.16**	**9.95**
DNN	3.83	5.03	5.54	5.83	5.52	7.63	10.02	10.45	10.83	10.55
LSTM	**3.81**	**4.46**	**4.83**	5.89	5.64	8.26	10.53	11.23	11.96	11.64
	OoR^2					MCPDC				
	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$
NTK	87.21	80.44	73.79	**72.81**	**76.59**	**45.33**	**59.56**	**67.26**	**72.63**	**76.65**
DNN	87.09	77.39	73.20	70.69	73.74	44.78	56.47	65.58	69.73	73.41
LSTM	**87.32**	**82.70**	**79.77**	70.11	72.56	45.17	56.14	64.96	69.61	72.50

Table A.8: Prediction accuracy in terms of RMSE, MAPE, OoR^2, and MCPDC when we use interpolated implied volatility values, with the NTK parameterized NN, standard NN, and LTSM as forecasting models. The prediction period is from Jan 09, 2019 to Dec 31, 2021. Bold numbers indicate the best-performing model (or models) in a given column.

toencoder (AE). However, instead of using the AHBS model in our implementation, we interpolate with the basis splines functions to obtain a better fit of IVS. The AHBS model is essentially a set of polynomial basis functions in terms of moneyness and time-to-maturity, which are less flexible than the splines functions (Ramsay and Silverman, 2005). Additionally, we have a much higher number of observed IV values, i.e., about 908 call options and 875 put options daily, compared to 374 options per day in Zhang, Li, and Zhang (2023). Hence, we use a larger grid of 483 points compared to the 154-point grid in Zhang, Li, and Zhang (2023). The interpolated values are obtained in the same fashion as in Section A.9.3. More details on how we use extract latent features with LSTM can be found in Appendix A.8.4.

In order to perform a fair comparison between PCA and AE, the number of eigenscores for PCA and the number of latent features for AE are kept to be equal. Specifically, we determine the number of eigenvectors for PCA, and match the number of latent features of AE to this number. This can be done using two commonly used hard cutoffs for PCA: the first eigenvector and first three eigenvectors, as well as using a threshold for variance explained: 90%, 95%, and 99%, which corresponds to the first 2, 4, and 8 eigenvectors. The results reported in Table A.9 show that the prediction accuracy of PCA improves monotonically as the number of eigenscores (features) increases. For all forecasting horizons, using the first 3 eigenscores for PCA leads to worse accuracy than using AE with 8 latent features, which is consistent with results in Zhang, Li, and Zhang (2023). However, if both PCA and AE have 8 extracted features, then PCA tends to outperform AE, e.g., for $h = 20$, RMSE for PCA is 7.88% while it is 8.97% for AE. Both dimension reduction methods, PCA and AE, give worse results than if we use the interpolated points without reducing their dimension, which is also in line with Zhang, Li, and Zhang (2023).

	Principal component analysis				Autoencoder			
# features	RMSE	MAPE	OoR^2	MCPDC	RMSE	MAPE	OoR^2	MCPDC
$h = 1$								
1	5.00	11.75	77.87	48.95	4.51	10.06	82.11	**49.38**
2	4.21	10.21	84.46	49.34	3.84	8.68	87.12	48.80
3	4.22	9.65	84.23	**49.93**	**3.54**	8.26	**88.99**	47.76
4	4.18	9.27	84.61	47.67	3.57	**8.09**	88.73	48.17
8	**3.88**	**7.74**	**86.82**	48.19	3.65	8.21	88.27	47.62
$h = 5$								
1	6.27	14.02	65.68	46.88	6.11	12.87	67.31	47.00
2	5.55	12.82	73.21	48.28	5.45	11.68	74.18	48.67
3	5.42	12.38	74.31	49.44	5.07	11.73	77.61	50.53
4	5.24	12.22	75.97	49.51	4.85	**11.19**	79.50	50.16
8	**4.52**	**10.47**	**82.19**	**54.12**	**4.80**	11.70	**79.98**	**51.01**
$h = 10$								
1	7.75	15.77	47.80	49.19	7.88	15.36	46.03	51.14
2	7.12	14.77	56.00	50.29	7.09	14.64	56.37	52.68
3	6.97	14.30	57.77	52.24	6.87	14.40	59.07	53.39
4	6.47	13.95	63.72	54.56	6.71	14.01	61.00	**54.33**
8	**5.80**	**12.80**	**70.70**	**58.31**	**6.20**	**13.69**	**66.51**	54.01
$h = 15$								
1	9.03	17.18	29.48	51.01	9.16	16.67	27.50	52.33
2	8.64	16.26	35.56	51.74	8.65	16.17	35.48	52.25
3	8.46	15.85	38.18	54.24	8.34	15.49	39.91	54.80
4	7.94	15.15	45.57	55.71	8.00	15.07	44.61	57.03
8	**6.79**	**13.57**	**60.04**	**60.31**	**7.75**	**14.60**	**48.08**	**58.76**
$h = 20$								
1	9.84	18.09	16.56	52.53	9.95	19.04	14.47	55.69
2	9.67	17.50	19.48	52.43	9.69	17.54	19.08	54.22
3	9.52	16.76	21.83	56.18	9.45	17.51	22.87	54.69
4	9.04	15.98	29.51	58.97	9.46	17.24	22.76	55.39
8	**7.88**	**14.40**	**46.29**	**62.32**	**8.97**	**16.15**	**30.44**	**59.46**

Table A.9: Prediction accuracy of models using principal component analysis (PCA) or autoencoder (AE) for dimension reduction and LSTM for forecasting extracted features in terms of RMSE, MAPE, OoR2, and MCPDC, across forecasting horizons $h = 1, 5, 10, 15$ and 20. The prediction period is from Jan 09, 2019 to Dec 31, 2021. Bold numbers indicate the best-performing model (or models) in a given column.

A.10 Delta-hedging of call and put options portfolios

Short call delta-hedging. In a short call delta-hedging strategy, we sell one call option contract hedged by a long position in delta shares of S&P 500. Our setup relies on trading signals extracted from the predicted IVS and includes only non-zero volume call options at time i. Let Q_i^{sc} be the set of call options whose implied volatilities are predicted to decrease on day $i+h$, or in other words, the set of call options to be sold on day i. The initial investment cost for such a delta-hedged portfolio is $\sum_{j\in Q_i^{sc}}(\Delta_{j,i}^{sc}S_i - C_{j,i})$, where $C_{j,i}$ and $\Delta_{j,i}^{sc} \in [0,1]$ denote the price and the Black-Scholes delta of the call option $j \in Q_i^{sc}$, and S_i is the closing S&P 500 stock price on the day i. The capital required for the portfolio is always positive. To reduce transaction costs, we hold the position for an h-day period without rebalancing the delta-hedges, as described in Goyal and Saretto (2009). The payoff on day $i+h$ is $\sum_{j\in Q_i^{sc}}(\Delta_{j,i}^{sc}S_{i+h} - C_{j,i+h})$. Thus, we have the return R_i^{sc} of the short call delta-hedging portfolio

$$R_i^{sc} = \frac{\sum_{j\in Q_i^{sc}}(\Delta_{j,i}^{sc}S_{i+h} - C_{j,i+h})}{\sum_{j\in Q_i^{sc}}(\Delta_{j,i}^{sc}S_i - C_{j,i})} - 1$$

Short put delta-hedging. The short put delta-hedging strategy involves selling one contract of a put option and delta-hedging the position by shorting the S&P 500 index. Like the call delta-hedging, we use the predicted IVS to extract trading signals and discretely observed put options to build a portfolio. Denote by Q_i^{sp} the set of put options to be sold on the day i, i.e., those put options with IV predicted to decrease on the day $i+h$. On day i, we have a cash inflow of $\sum_{j\in Q_i^{sp}}(-\Delta_{j,i}^{sp}S_i + P_{j,i})$ where $P_{j,i}$ and $\Delta_{j,i}^{sp} \in [-1,0]$ denote the price and the Black-Scholes delta of the put option $j \in Q_i^{p}$. We hold the portfolio for h days without rebalancing, and on day $i+h$, we close the positions and pay a cost of $\sum_{j\in Q_i^{sp}}(-\Delta_{j,i}^{sp}S_{i+h} + P_{j,i+h})$. The returns of the portfolio is

$$R_i^{sp} = 1 - \frac{\sum_{j\in Q_i^{sp}}(-\Delta_{j,i}^{sp}S_{i+h} + P_{j,i+h})}{\sum_{j\in Q_i^{sp}}(-\Delta_{j,i}^{sp}S_i + P_{j,i})}$$

A.11 Additional results on economic performance

This section reports formal statistical tests on the Sharpe ratio differences in Section 5.2. We use the bootstrap inference for time series data to determine whether two Sharpe ratios of two models are significantly different, see Ledoit and Wolf (2008). Since the block size b of the bootstrap is data dependent, Algorithm 3.1 of Ledoit and Wolf (2008) is utilized to calibrate and choose the best block size b from $\{5, 10, 20, 25, 30, 35, 40, 45, 50\}$. Furthermore, as the test requires the time series to be of the same length, for a model j, if there is no trading signal from this model on some day i, we assume the excess return of model j on the day i is 0. The p-values of the bootstrap tests are shown in Table A.10.

Additionally, we also detail the performance of the remaining short strategies. Overall, fNTK performs the best over different strategies and trading periods, transaction costs, and filtering thresholds. But some differences exist. We review the main findings below.

The straddle strategies show better performance regarding mean simple returns and Sharpe ratio than delta-hedging strategies. Additionally, it is worth noting that theoretically, delta-hedging (with put and call together), simple straddle (using one call and one put option), and delta-neutral straddle (described in the main text) strategies using at-the-money (ATM) options, should yield the same returns. We verified this by using call and put ATM options with $0.48 \leq |\Delta| \leq 0.52$, and our results show that delta-hedging, simple straddle, and delta-neutral strategies with ATM options yield highly similar results, see Table A.16.

Short call delta-hedging strategy, Table A.11 and Figure A.4 is the only one that does not fully benefit from the nonlinear functional models and has out-performance in the benchmark models. In particular, the best-performing model is fRW which records high Sharpe ratios between 4.74 and 5.19 in 2021, while fNTK records a Sharpe ratio of 2.22 to 4.24 in the same period. However, this changes when we increase the filtering threshold to 10%; then fRW performs worse and fNTK performs the best in the longer horizons, e.g., for 20 steps ahead, as shown in Figure A.6. fNTK scores highest in short-put delta-hedging strategies, Table A.12 and Figure A.5, but the returns are

	DNN-RW	CW-RW	CW-DNN	AHBS-RW	AHBS-DNN	AHBS-VAR	AE-LSTM	fRW	fLinK	fGauK	fLapK	fNTK
$h = 1$												
DNN-RW	-	0.013	0.004	0.006	0.094	0.002	0.891	0.009	0.104	0.144	0.371	0.012
CW-RW	0.013	-	0.214	0.001	0.001	0.001	0.009	0.812	0.437	0.404	0.145	0.881
CW-DNN	0.004	0.214	-	0.001	0.001	0.001	0.004	0.758	0.212	0.203	0.061	0.522
AHBS-RW	0.006	0.001	0.001	-	0.001	0.168	0.455	0.001	0.003	0.001	0.007	0.001
AHBS-DNN	0.094	0.001	0.001	0.001	-	0.002	0.037	0.001	0.009	0.009	0.038	0.002
AHBS-VAR	0.002	0.001	0.001	0.168	0.002	-	0.301	0.001	0.001	0.002	0.001	0.001
AE-LSTM	0.891	0.009	0.004	0.455	0.037	0.301	-	0.010	0.111	0.138	0.323	0.017
fRW	0.009	0.812	0.758	0.001	0.001	0.001	0.010	-	0.230	0.183	0.059	0.660
fLinK	0.104	0.437	0.212	0.003	0.009	0.001	0.111	0.230	-	0.945	0.445	0.470
fGauK	0.144	0.404	0.203	0.001	0.009	0.002	0.138	0.183	0.945	-	0.421	0.335
fLapK	0.371	0.145	0.061	0.007	0.038	0.001	0.323	0.059	0.445	0.421	-	0.123
fNTK	0.012	0.881	0.522	0.001	0.002	0.001	0.017	0.660	0.470	0.335	0.123	-
$h = 5$												
DNN-RW	-	0.095	0.622	0.022	0.035	0.009	0.298	0.061	0.105	0.538	0.480	0.095
CW-RW	0.095	-	0.064	0.093	0.291	0.075	0.685	0.008	0.874	0.049	0.067	0.005
CW-DNN	0.622	0.064	-	0.037	0.056	0.036	0.664	0.007	0.221	0.278	0.273	0.020
AHBS-RW	0.022	0.093	0.037	-	0.087	0.850	0.086	0.009	0.177	0.009	0.008	0.003
AHBS-DNN	0.035	0.291	0.056	0.087	-	0.278	0.028	0.010	0.397	0.012	0.007	0.004
AHBS-VAR	0.009	0.075	0.036	0.850	0.278	-	0.880	0.004	0.081	0.007	0.012	0.003
AE-LSTM	0.298	0.685	0.664	0.086	0.028	0.880	-	0.012	0.501	0.210	0.156	0.065
fRW	0.061	0.008	0.007	0.009	0.010	0.004	0.012	-	0.015	0.051	0.046	0.193
fLinK	0.105	0.874	0.221	0.177	0.397	0.081	0.501	0.015	-	0.030	0.042	0.007
fGauK	0.538	0.049	0.278	0.009	0.012	0.007	0.210	0.051	0.030	-	0.974	0.215
fLapK	0.480	0.067	0.273	0.008	0.007	0.012	0.156	0.046	0.042	0.974	-	0.139
fNTK	0.095	0.005	0.020	0.003	0.004	0.003	0.065	0.193	0.007	0.215	0.139	-
$h = 10$												
DNN-RW	-	0.037	0.421	0.047	0.069	0.085	0.047	0.011	0.601	0.017	0.006	0.001
CW-RW	0.037	-	0.020	0.768	0.231	0.854	0.021	0.007	0.328	0.005	0.006	0.001
CW-DNN	0.421	0.020	-	0.035	0.062	0.060	0.164	0.005	0.305	0.020	0.012	0.001
AHBS-RW	0.047	0.768	0.035	-	0.093	0.983	0.099	0.017	0.411	0.009	0.002	0.001
AHBS-DNN	0.069	0.231	0.062	0.093	-	0.510	0.038	0.015	0.710	0.010	0.001	0.001
AHBS-VAR	0.085	0.854	0.060	0.983	0.510	-	0.120	0.011	0.327	0.011	0.004	0.001
AE-LSTM	0.047	0.021	0.164	0.099	0.038	0.120	-	0.019	0.043	0.293	0.022	0.003
fRW	0.011	0.007	0.005	0.017	0.015	0.011	0.019	-	0.016	0.421	0.531	0.056
fLinK	0.601	0.328	0.305	0.411	0.710	0.327	0.043	0.016	-	0.013	0.002	0.001
fGauK	0.017	0.005	0.020	0.009	0.010	0.011	0.293	0.421	0.013	-	0.392	0.006
fLapK	0.006	0.006	0.012	0.002	0.001	0.004	0.022	0.531	0.002	0.392	-	0.136
fNTK	0.001	0.001	0.001	0.001	0.001	0.001	0.003	0.056	0.001	0.006	0.136	-
$h = 15$												
DNN-RW	-	0.037	0.078	0.300	0.672	0.305	0.040	0.023	0.533	0.024	0.018	0.004
CW-RW	0.037	-	0.057	0.087	0.062	0.946	0.015	0.011	0.534	0.011	0.015	0.001
CW-DNN	0.078	0.057	-	0.046	0.071	0.033	0.210	0.022	0.132	0.105	0.026	0.004
AHBS-RW	0.300	0.087	0.046	-	0.249	0.350	0.250	0.032	0.861	0.028	0.016	0.001
AHBS-DNN	0.672	0.062	0.071	0.249	-	0.507	0.102	0.025	0.710	0.034	0.021	0.004
AHBS-VAR	0.305	0.946	0.033	0.350	0.507	-	0.079	0.033	0.579	0.031	0.015	0.001
AE-LSTM	0.040	0.015	0.210	0.250	0.102	0.079	-	0.254	0.043	0.264	0.041	0.035
fRW	0.023	0.011	0.022	0.032	0.025	0.033	0.254	-	0.040	0.813	0.266	0.108
fLinK	0.533	0.534	0.132	0.861	0.710	0.579	0.043	0.040	-	0.015	0.010	0.001
fGauK	0.024	0.011	0.105	0.028	0.034	0.031	0.264	0.813	0.015	-	0.168	0.074
fLapK	0.018	0.015	0.026	0.016	0.021	0.015	0.041	0.266	0.010	0.168	-	0.155
fNTK	0.004	0.001	0.004	0.001	0.004	0.001	0.035	0.108	0.001	0.074	0.155	-
$h = 20$												
DNN-RW	-	0.769	0.069	0.576	0.944	0.765	0.101	0.113	0.770	0.050	0.011	0.003
CW-RW	0.769	-	0.112	0.966	0.767	0.874	0.182	0.140	0.712	0.055	0.006	0.006
CW-DNN	0.069	0.112	-	0.078	0.086	0.270	0.896	0.170	0.297	0.146	0.078	0.040
AHBS-RW	0.576	0.966	0.078	-	0.401	0.881	0.161	0.134	0.610	0.043	0.009	0.003
AHBS-DNN	0.944	0.767	0.086	0.401	-	0.739	0.141	0.127	0.803	0.043	0.017	0.003
AHBS-VAR	0.765	0.874	0.270	0.881	0.739	-	0.172	0.147	0.463	0.079	0.007	0.007
AE-LSTM	0.101	0.182	0.896	0.161	0.141	0.172	-	0.223	0.238	0.077	0.054	0.041
fRW	0.113	0.140	0.170	0.134	0.127	0.147	0.223	-	0.148	0.554	0.178	0.092
fLinK	0.770	0.712	0.297	0.610	0.803	0.463	0.238	0.148	-	0.083	0.011	0.006
fGauK	0.050	0.055	0.146	0.043	0.043	0.079	0.077	0.554	0.083	-	0.216	0.082
fLapK	0.011	0.006	0.078	0.009	0.017	0.007	0.054	0.178	0.011	0.216	-	0.008
fNTK	0.003	0.006	0.040	0.003	0.003	0.007	0.041	0.092	0.006	0.082	0.008	-

Table A.10: P-values for statistical tests for Sharpe ratios reported in Section 5.2. The hypothesis being tested is $H_0 : SR_i = SR_j$ against an alternative $H_1 : SR_i \neq SR_j$, where model i is the label of the selected row, whereas model j is the label of the selected column. For each forecasting horizon h, if the reported p-value at the i^{th} row and j^{th} column is less than 0.05, then the model i performs statistically different from model j at 5% level of significance.

close to zero and negative for most horizons.

By looking at the trading returns across different years in Tables A.11 and A.12, we observe a reasonable pattern: for short strategies, the returns are generally better in the calmer periods of years 2019 and 2021. Furthermore, across all trading strategies, it is also noteworthy that as we increase the filtering threshold, the nonlinear models, especially the fNTK model, tend to have better returns; though it is not always true for the classical methods, peculiarly the fRW model. For transaction costs, as the effective spread measures, i.e., the trading costs become higher, the returns of the models tend to reduce, but not by a significant amount.

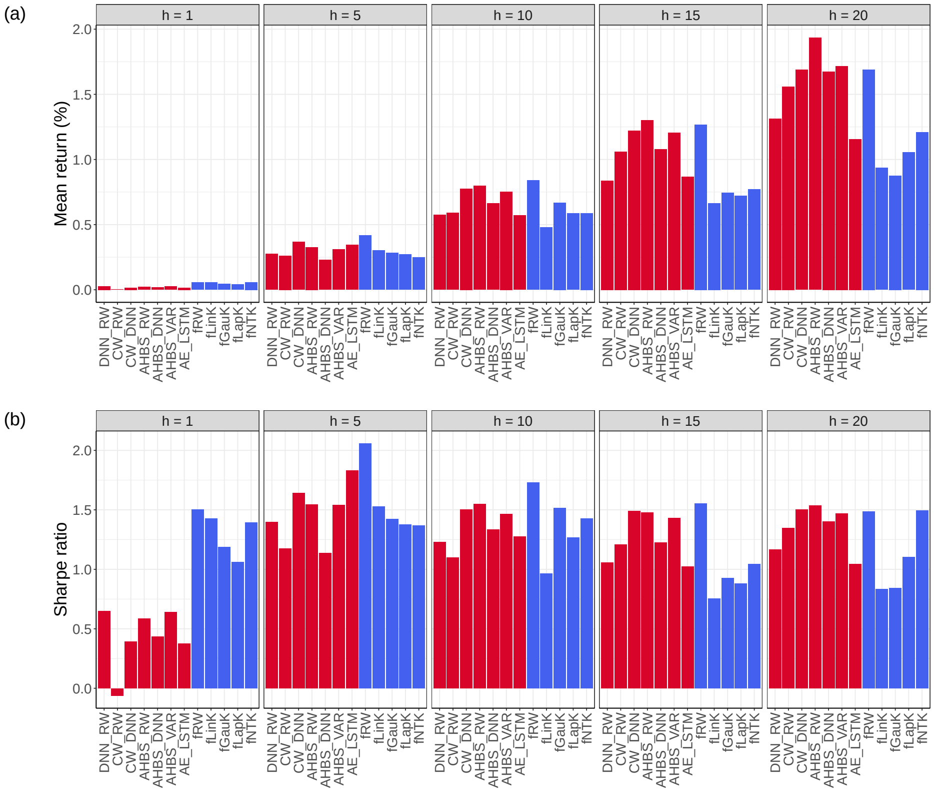


Figure A.4: Mean simple returns (MR) in percentage and Sharpe ratio (SR) of short call delta-hedging strategy. The prediction period is from Jan 09, 2019 to Dec 31, 2021.

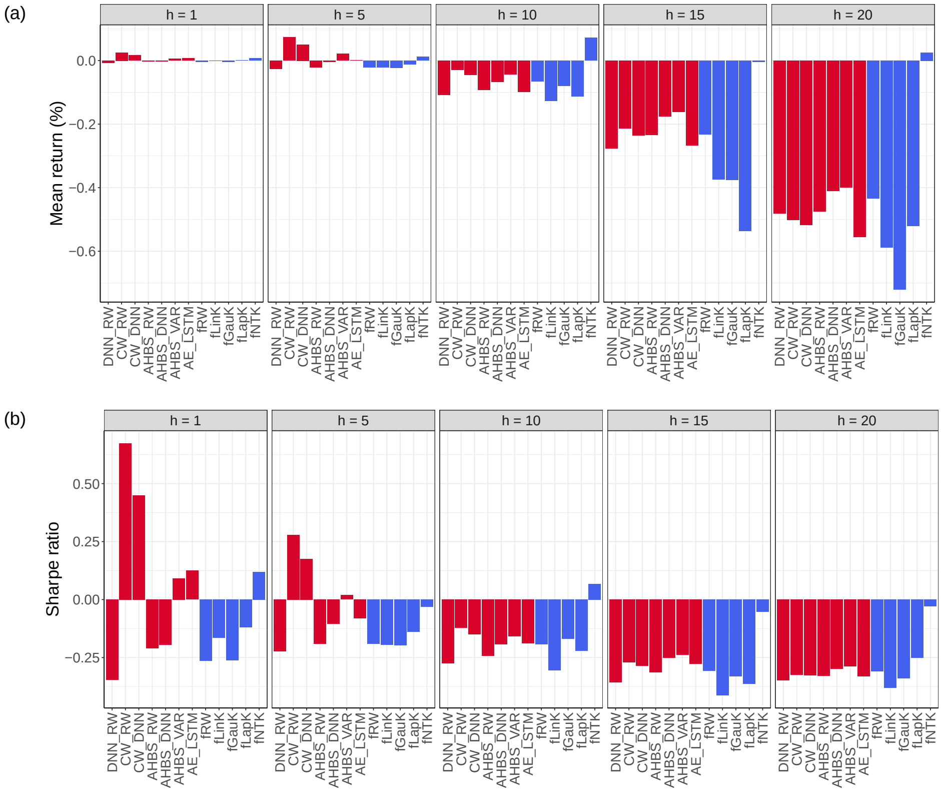


Figure A.5: Mean simple returns (MR) in percentage and annualized Sharpe ratio of short put delta-hedging strategy. The prediction period is from Jan 09, 2019 to Dec 31, 2021.

	Mean return (%)					Sharpe ratio				
	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$
Overall (from Jan 9, 2019 to Dec 31, 2021)										
DNN-RW	0.03	0.27	0.57	0.84	1.31	0.65	1.40	1.23	1.06	1.17
CW-RW	0.00	0.26	0.59	1.06	1.56	-0.06	1.18	1.10	1.21	1.35
CW-DNN	0.02	0.37	0.78	1.22	1.69	0.39	1.64	1.50	1.49	1.50
AHBS-RW	0.02	0.33	0.80	**1.30**	**1.93**	0.59	1.54	1.55	1.48	**1.54**
AHBS-DNN	0.02	0.23	0.66	1.08	1.68	0.43	1.14	1.33	1.23	1.40
AHBS-VAR	0.02	0.31	0.75	1.21	1.72	0.64	1.54	1.47	1.43	1.47
AE-LSTM	0.02	0.34	0.57	0.87	1.16	0.38	1.83	1.27	1.02	1.05
fRW	**0.06**	**0.42**	**0.84**	1.27	1.69	**1.51**	**2.06**	**1.73**	**1.55**	1.49
fLinK	0.06	0.30	0.48	0.66	0.94	1.43	1.53	0.97	0.76	0.83
fGauK	0.04	0.28	0.67	0.74	0.88	1.19	1.42	1.52	0.93	0.84
fLapK	0.04	0.27	0.59	0.72	1.05	1.06	1.38	1.27	0.88	1.10
fNTK	0.06	0.25	0.59	0.77	1.21	1.39	1.37	1.43	1.04	1.50
From Jan 9, 2019 to Dec 31, 2019										
DNN-RW	0.03	0.29	0.70	1.01	1.57	1.93	2.27	2.80	2.55	2.69
CW-RW	0.01	0.22	0.53	0.86	1.45	0.53	2.10	2.40	2.33	2.64
CW-DNN	0.03	0.37	0.81	1.24	1.71	1.36	3.12	3.46	3.47	3.32
AHBS-RW	0.03	0.34	0.93	1.53	2.03	1.65	3.04	3.71	3.74	3.55
AHBS-DNN	0.02	0.27	0.81	1.40	1.95	1.19	2.58	3.29	3.33	3.22
AHBS-VAR	0.03	0.33	**1.08**	**1.71**	**2.34**	1.55	3.01	**4.41**	**4.13**	**4.25**
AE-LSTM	0.03	0.36	0.67	1.31	1.42	1.52	3.17	3.26	3.59	2.98
fRW	0.05	**0.45**	0.94	1.40	1.86	2.99	**3.95**	4.19	3.95	3.62
fLinK	0.06	0.34	0.60	0.74	0.81	3.24	3.17	3.19	2.17	2.01
fGauK	**0.07**	0.41	0.83	1.18	1.31	**3.72**	3.93	4.23	3.49	3.08
fLapK	0.06	0.39	0.82	1.14	1.39	3.55	3.87	4.55	3.18	3.45
fNTK	0.05	0.27	0.67	0.89	1.18	2.81	2.64	3.96	2.82	3.26
From Jan 1, 2020 to Dec 31, 2020										
DNN-RW	-0.01	0.17	0.30	0.37	0.78	-0.19	0.59	0.41	0.30	0.44
CW-RW	-0.03	**0.37**	**0.76**	**1.40**	**1.85**	-0.64	1.06	**0.90**	**1.02**	1.03
CW-DNN	-0.02	**0.37**	0.71	1.18	1.70	-0.38	**1.08**	0.87	0.91	0.96
AHBS-RW	-0.01	0.18	0.42	0.80	1.68	-0.31	0.56	0.52	0.57	0.85
AHBS-DNN	-0.01	0.02	0.19	0.33	1.08	-0.31	0.04	0.23	0.23	0.58
AHBS-VAR	0.00	0.21	0.34	0.49	0.93	-0.09	0.69	0.42	0.37	0.52
AE-LSTM	-0.02	0.21	0.23	-0.12	0.65	-0.32	0.77	0.33	-0.12	0.38
fRW	0.02	0.25	0.55	0.88	1.21	0.26	0.80	0.72	0.68	0.68
fLinK	**0.03**	0.11	0.03	-0.04	0.43	**0.40**	0.37	0.02	-0.05	0.24
fGauK	-0.02	0.01	0.35	0.03	0.05	-0.38	0.01	0.50	0.00	0.00
fLapK	-0.02	0.02	0.16	-0.09	0.23	-0.33	0.04	0.21	-0.10	0.14
fNTK	0.05	0.13	0.44	0.46	1.26	0.72	0.46	0.68	0.40	**1.04**
From Jan 1, 2021 to Dec 31, 2021										
DNN-RW	0.06	0.37	0.73	1.15	1.59	3.12	3.21	3.24	3.24	3.20
CW-RW	0.02	0.19	0.47	0.91	1.36	1.49	2.73	2.84	2.98	2.95
CW-DNN	0.04	0.36	0.81	1.25	1.65	2.52	3.81	3.99	3.83	3.64
AHBS-RW	0.05	0.46	**1.06**	**1.61**	**2.11**	3.23	4.16	4.40	4.38	4.34
AHBS-DNN	0.04	0.40	1.01	1.54	2.03	2.89	3.88	4.14	4.17	4.00
AHBS-VAR	0.05	0.40	0.85	1.44	1.92	3.03	3.58	3.71	3.64	3.46
AE-LSTM	0.04	0.46	0.82	1.47	1.43	2.52	4.03	3.73	4.82	2.38
fRW	**0.10**	**0.56**	1.05	1.53	2.02	**4.92**	**5.19**	**5.08**	**4.80**	**4.74**
fLinK	0.09	0.45	0.82	1.35	1.65	4.60	4.41	3.90	3.92	3.33
fGauK	0.09	0.44	0.83	1.01	1.25	4.45	5.14	3.96	2.75	1.93
fLapK	0.08	0.41	0.79	1.12	1.55	3.99	4.97	4.18	3.68	3.20
fNTK	0.08	0.35	0.66	0.95	1.19	4.24	4.22	3.42	2.62	2.22

Table A.11: Mean simple returns (MR) and annualized Sharpe ratio (SR) of short call delta-hedging over the whole test period, from Jan 9, 2019 to Dec 31, 2021, and in each year of the test period. Bold numbers indicate the best-performing model (or models) in a given column.

	Mean return (%)					Sharpe ratio				
	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$
Overall (from Jan 9, 2019 to Dec 31, 2021)										
DNN-RW	-0.01	-0.03	-0.11	-0.28	-0.48	-0.35	-0.22	-0.28	-0.36	-0.35
CW-RW	**0.03**	**0.07**	-0.03	-0.21	-0.50	**0.67**	**0.28**	-0.12	-0.27	-0.32
CW-DNN	0.02	0.05	-0.04	-0.24	-0.52	0.45	0.17	-0.15	-0.29	-0.33
AHBS-RW	0.00	-0.02	-0.09	-0.23	-0.47	-0.21	-0.19	-0.24	-0.31	-0.33
AHBS-DNN	0.00	0.00	-0.07	-0.18	-0.41	-0.20	-0.11	-0.19	-0.25	-0.30
AHBS-VAR	0.01	0.02	-0.04	-0.16	-0.40	0.09	0.02	-0.16	-0.24	-0.29
AE-LSTM	0.01	0.00	-0.10	-0.27	-0.56	0.12	-0.08	-0.19	-0.28	-0.33
fRW	0.00	-0.02	-0.07	-0.23	-0.43	-0.26	-0.19	-0.19	-0.31	-0.31
fLinK	0.00	-0.02	-0.13	-0.37	-0.59	-0.16	-0.20	-0.31	-0.41	-0.38
fGauK	0.00	-0.02	-0.08	-0.38	-0.72	-0.26	-0.20	-0.17	-0.33	-0.34
fLapK	0.00	-0.01	-0.11	-0.54	-0.52	-0.12	-0.14	-0.22	-0.36	-0.25
fNTK	0.01	0.01	**0.07**	**0.00**	**0.03**	0.12	-0.03	**0.07**	**-0.05**	**-0.03**
From Jan 9, 2019 to Dec 31, 2019										
DNN-RW	0.01	0.06	0.12	0.17	0.21	0.03	0.17	0.25	0.24	0.15
CW-RW	**0.03**	**0.13**	0.17	0.18	0.19	**1.51**	**1.12**	0.64	0.30	0.06
CW-DNN	0.02	0.11	0.17	0.17	0.18	1.07	0.94	0.64	0.23	0.00
AHBS-RW	0.01	0.06	0.11	0.16	0.21	0.35	0.21	0.17	0.14	0.13
AHBS-DNN	0.01	0.07	0.13	0.19	0.23	0.29	0.36	0.38	0.37	0.25
AHBS-VAR	0.02	0.08	0.14	0.23	**0.32**	1.01	0.46	0.44	**0.59**	**0.66**
AE-LSTM	0.02	0.09	**0.21**	**0.25**	0.27	0.52	0.62	**0.91**	0.63	0.37
fRW	0.01	0.06	0.12	0.16	0.23	0.11	0.21	0.28	0.18	0.24
fLinK	0.01	0.06	0.16	0.24	0.28	0.39	0.22	0.59	0.64	0.46
fGauK	0.01	0.06	0.11	0.16	0.21	0.16	0.18	0.18	0.11	0.12
fLapK	0.01	0.05	0.14	0.23	0.24	0.40	0.04	0.41	0.53	0.24
fNTK	0.01	0.07	0.16	0.22	0.24	0.27	0.32	0.60	0.50	0.28
From Jan 1, 2020 to Dec 31, 2020										
DNN-RW	-0.04	-0.24	-0.70	-1.35	-2.12	-0.88	-0.78	-0.84	-0.92	-0.83
CW-RW	**-0.01**	**-0.13**	-0.62	-1.24	-2.24	**-0.13**	**-0.42**	-0.71	-0.77	-0.77
CW-DNN	**-0.01**	-0.16	-0.67	-1.33	-2.35	-0.32	-0.53	-0.77	-0.81	-0.79
AHBS-RW	-0.03	-0.23	-0.70	-1.29	-2.28	-0.74	-0.74	-0.83	-0.89	-0.86
AHBS-DNN	-0.04	-0.21	-0.66	-1.18	-2.05	-0.83	-0.67	-0.78	-0.82	-0.80
AHBS-VAR	-0.03	-0.14	-0.55	-1.02	-1.73	-0.71	-0.45	-0.69	-0.72	-0.69
AE-LSTM	-0.03	-0.16	-0.78	-1.38	-2.03	-0.69	-0.48	-0.68	-0.75	-0.70
fRW	-0.04	-0.27	-0.65	-1.31	-2.14	-0.95	-0.87	-0.78	-0.88	-0.81
fLinK	-0.04	-0.24	-0.79	-1.67	-2.31	-0.94	-0.78	-0.94	-1.00	-0.84
fGauK	-0.04	-0.27	-0.65	-1.53	-2.59	-1.09	-0.82	-0.60	-0.76	-0.73
fLapK	-0.04	-0.21	-0.77	-2.00	-2.09	-1.04	-0.69	-0.74	-0.79	-0.57
fNTK	-0.02	-0.19	**-0.33**	**-0.65**	**-0.81**	-0.60	-0.57	**-0.41**	**-0.38**	**-0.34**
From Jan 1, 2021 to Dec 31, 2021										
DNN-RW	0.01	0.11	0.29	0.45	0.60	0.59	1.87	2.38	2.56	2.42
CW-RW	**0.05**	**0.22**	0.37	0.46	0.62	**2.78**	**3.05**	2.81	2.21	2.08
CW-DNN	0.04	0.20	0.38	0.51	0.69	2.40	3.03	3.19	2.99	3.07
AHBS-RW	0.01	0.12	0.34	0.53	0.78	0.78	2.01	2.82	3.13	3.38
AHBS-DNN	0.02	0.14	0.36	0.56	0.78	1.11	2.27	3.05	3.41	3.40
AHBS-VAR	0.03	0.13	0.34	0.52	0.81	1.51	1.96	2.59	2.57	3.07
AE-LSTM	0.03	0.10	0.35	0.48	0.54	1.80	1.23	2.70	2.76	1.98
fRW	0.02	0.16	0.39	0.53	0.73	1.26	2.38	3.40	3.86	3.19
fLinK	0.02	0.15	0.35	0.57	0.87	1.26	2.14	2.59	2.87	3.81
fGauK	0.02	0.16	0.38	0.53	0.83	1.34	2.40	3.19	3.80	4.07
fLapK	0.03	0.16	**0.42**	**0.60**	**0.91**	1.63	2.31	**3.76**	**4.18**	**4.44**
fNTK	0.03	0.17	**0.42**	0.54	0.79	1.72	2.36	3.38	4.10	3.94

Table A.12: Mean simple returns in percentage and annualized Sharpe ratio of short put delta-hedging over the whole test period, from Jan 9, 2019 to Dec 31, 2021, and in each year of the test period. Bold numbers indicate the best-performing model (or models) in a given column.

	Mean return (%)					Sharpe ratio				
	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$
Overall (from Jan 9, 2019 to Dec 31, 2021)										
DNN-RW	-0.05	1.03	0.07	-0.79	-2.54	-0.14	0.37	0.01	-0.09	-0.16
CW-RW	0.21	0.36	-1.10	-2.44	-2.34	0.57	0.14	-0.21	-0.28	-0.19
CW-DNN	**0.27**	0.83	0.58	0.56	0.90	**0.72**	0.31	0.09	0.05	0.06
AHBS-RW	0.03	1.69	4.31	4.39	4.62	0.06	0.61	0.66	0.40	0.26
AHBS-DNN	0.15	2.09	5.04	5.40	5.21	0.39	0.75	0.77	0.50	0.30
AHBS-VAR	-0.12	0.57	0.67	-0.96	-3.16	-0.33	0.20	0.10	-0.09	-0.18
AE-LSTM	-0.04	0.83	2.54	2.76	0.89	-0.11	0.26	0.36	0.23	0.04
fRW	**0.27**	**2.90**	5.89	4.92	4.17	0.67	**0.99**	0.89	0.44	0.24
fLinK	0.16	0.43	-0.51	-2.51	-2.52	0.37	0.14	-0.08	-0.22	-0.15
fGauK	0.15	1.68	4.27	4.43	4.80	0.35	0.55	0.70	0.40	0.31
fLapK	0.07	1.64	6.44	6.79	9.88	0.16	0.54	1.12	0.67	0.99
fNTK	0.23	2.20	**7.49**	**9.47**	**14.40**	0.56	0.74	**1.70**	**1.30**	**1.83**
From Jan 9, 2019 to Dec 31, 2019										
DNN-RW	0.50	2.35	3.06	4.13	5.13	1.40	1.14	1.06	1.21	1.13
CW-RW	0.33	0.81	1.21	1.86	3.72	1.13	0.51	0.51	0.64	0.89
CW-DNN	0.53	1.82	3.78	6.43	6.80	1.63	1.04	1.34	1.84	1.51
AHBS-RW	0.48	2.47	6.45	8.70	10.45	1.49	1.29	1.93	1.97	1.83
AHBS-DNN	0.57	3.20	7.26	9.43	11.33	1.66	1.65	2.14	2.18	1.98
AHBS-VAR	0.15	1.68	6.94	7.42	12.09	0.49	0.82	1.83	1.64	2.03
AE-LSTM	0.50	3.07	8.82	9.55	10.73	1.46	1.23	2.51	2.24	1.91
fRW	**0.68**	**4.05**	**9.52**	9.91	10.89	**1.95**	**2.00**	2.62	2.27	1.99
fLinK	0.46	1.21	4.43	7.36	8.92	1.28	0.58	1.44	2.07	1.90
fGauK	0.41	1.78	4.45	8.37	9.93	1.14	0.83	1.38	2.03	2.29
fLapK	0.60	1.96	8.52	**11.71**	**11.82**	1.62	0.83	**2.77**	**3.12**	2.36
fNTK	0.52	1.80	7.93	9.17	11.60	1.41	0.77	2.60	2.58	**2.56**
From Jan 1, 2020 to Dec 31, 2020										
DNN-RW	-0.53	-1.57	-7.60	-16.07	-27.08	-1.13	-0.40	-0.82	-1.01	-1.00
CW-RW	**-0.03**	-1.89	-9.31	-17.49	-23.15	**-0.06**	-0.49	-1.01	-1.16	-1.08
CW-DNN	-0.13	-2.70	-9.75	-17.17	-22.35	-0.27	-0.68	-1.02	-1.09	-1.03
AHBS-RW	-0.52	-1.68	-5.48	-12.75	-23.49	-1.13	-0.43	-0.56	-0.78	-0.84
AHBS-DNN	-0.28	-2.10	-4.48	-11.31	-19.62	-0.60	-0.53	-0.48	-0.70	-0.74
AHBS-VAR	-0.56	-1.66	-6.93	-9.57	-16.12	-1.22	-0.43	-0.76	-0.64	-0.68
AE-LSTM	-0.70	-2.17	-5.76	-16.69	-18.46	-1.42	-0.52	-0.58	-0.98	-0.68
fRW	-0.41	-1.92	-6.26	-15.90	-26.73	-0.87	-0.45	-0.62	-0.91	-0.95
fLinK	-0.63	-2.93	-9.76	-19.93	-24.28	-1.30	-0.71	-0.98	-1.14	-0.91
fGauK	-0.63	-2.97	-3.89	-10.73	-15.55	-1.30	-0.70	-0.46	-0.71	-0.70
fLapK	-0.70	-1.89	-1.48	-6.10	-0.65	-1.46	-0.46	-0.19	-0.43	-0.05
fNTK	-0.36	**-1.05**	**2.90**	**0.46**	**8.22**	-0.75	**-0.25**	**0.47**	**0.04**	**0.75**
From Jan 1, 2021 to Dec 31, 2021										
DNN-RW	-0.10	2.41	5.17	8.58	10.56	-0.33	1.85	2.94	2.97	3.28
CW-RW	0.33	2.01	3.53	5.75	8.79	1.27	2.18	2.50	2.74	2.68
CW-DNN	0.40	3.27	7.16	10.34	15.58	1.40	3.07	4.19	4.55	4.77
AHBS-RW	0.15	4.52	12.52	18.45	25.57	0.44	3.07	4.53	4.30	5.19
AHBS-DNN	0.19	5.50	14.00	19.57	25.22	0.59	3.58	5.03	4.80	5.19
AHBS-VAR	0.07	2.08	6.03	8.00	8.73	0.22	1.79	2.64	2.69	3.02
AE-LSTM	0.08	2.91	7.58	21.82	20.54	0.24	1.86	2.80	5.23	3.03
fRW	0.59	6.70	14.26	18.45	**23.78**	1.71	**4.51**	**5.76**	5.00	**5.33**
fLinK	0.73	3.91	7.69	12.41	19.40	1.94	2.88	3.03	3.37	4.09
fGauK	**0.75**	**6.82**	14.00	19.86	23.68	**2.02**	4.20	4.80	3.44	4.23
fLapK	0.42	5.45	**14.91**	**20.12**	22.21	1.21	3.95	5.37	**5.35**	3.97
fNTK	0.61	6.01	12.20	19.26	21.82	1.70	4.29	4.74	4.75	3.60

Table A.13: Mean simple returns (%) and annualized Sharpe ratio of short delta-neutral straddles over the whole test period, from Jan 9, 2019 to Dec 31, 2021, and over each year (2019, 2020, and 2021) of the test period. Bold numbers indicate the best-performing model (or models) in a given column.

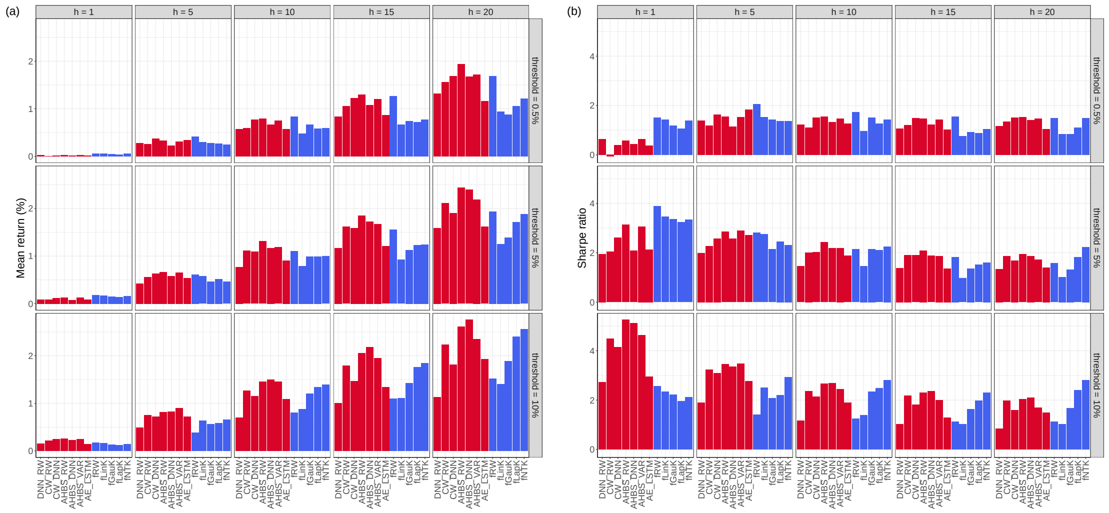


Figure A.6: Mean simple returns (MR) in percentage and annualized Sharpe ratio of short call delta-hedging strategy using three levels of filtering thresholds: 0.5%, 5% and 10%. The prediction period is from Jan 09, 2019 to Dec 31, 2021.

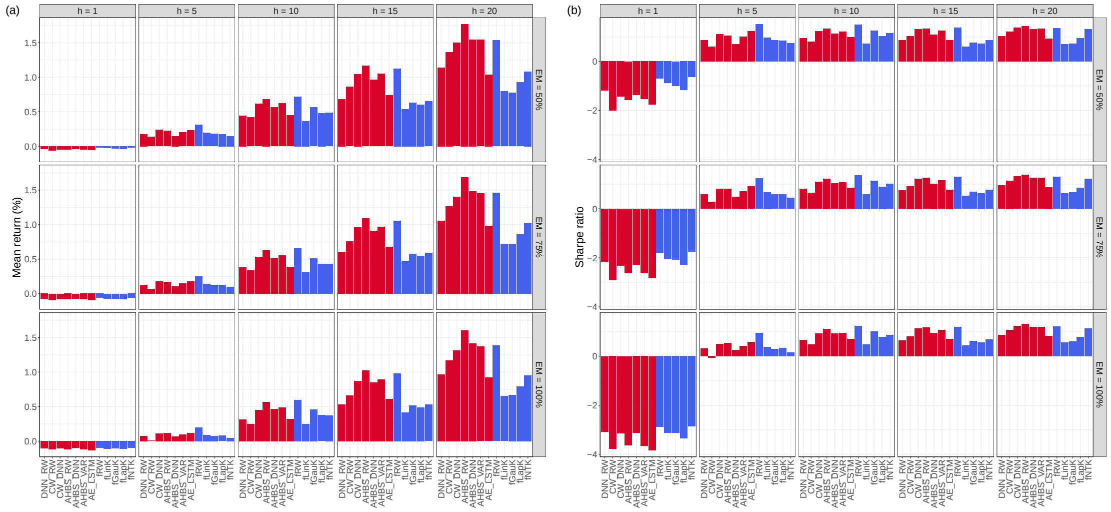


Figure A.7: Mean simple returns (MR) in percentage and annualized Sharpe ratio of short call delta-hedging strategy using three levels of effective measure (EM): 50%, 75% and 100%. The prediction period is from Jan 09, 2019 to Dec 31, 2021.

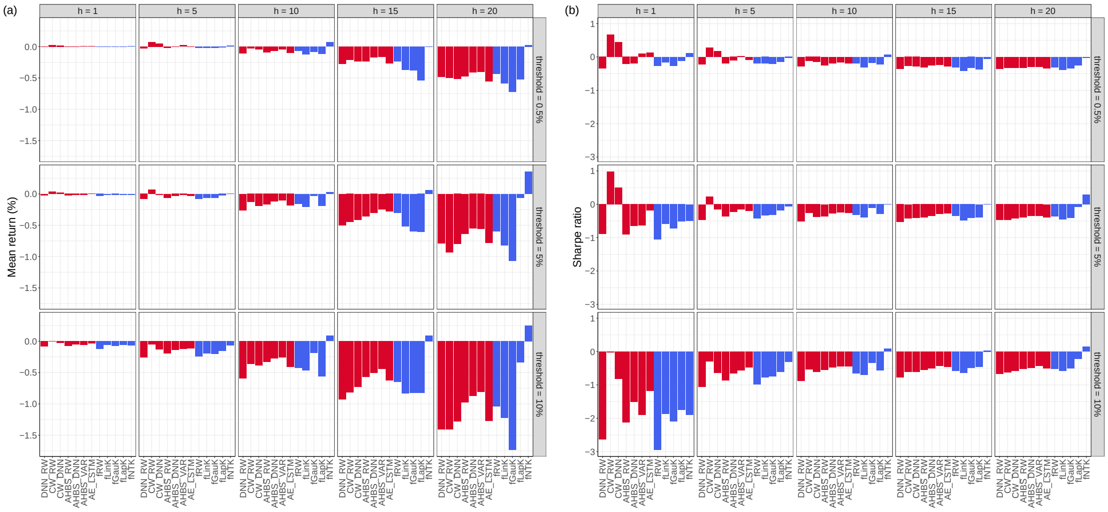


Figure A.8: Mean simple returns (%) and annualized Sharpe ratio of short put delta-hedging strategy using three levels of filtering thresholds: 0.5%, 5% and 10%. The prediction period is from Jan 09, 2019 to Dec 31, 2021.

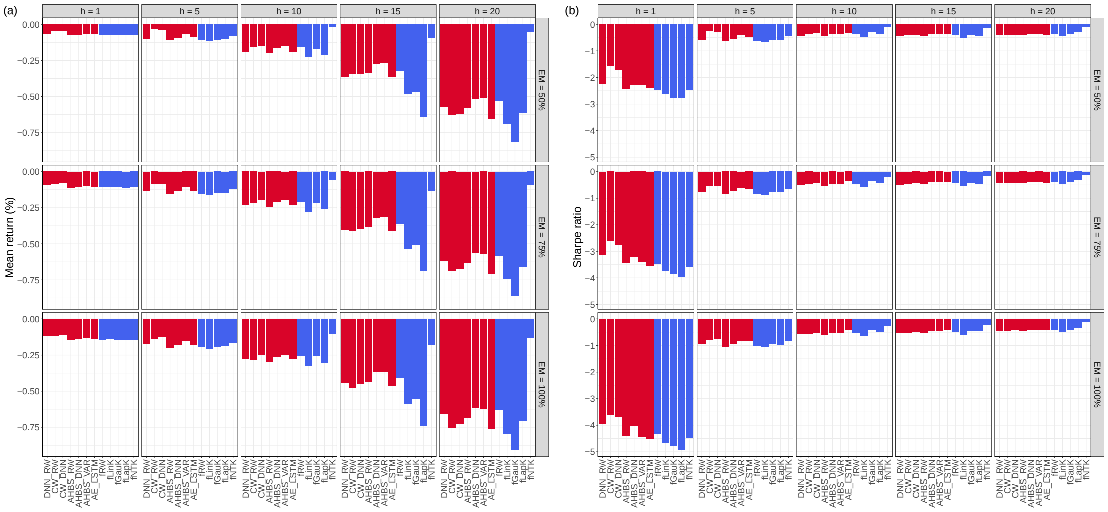


Figure A.9: Mean simple returns (%) and annualized Sharpe ratio of short put delta-hedging strategy using three levels of effective measure (EM): 50%, 75% and 100%. The prediction period is from Jan 09, 2019 to Dec 31, 2021.

	Mean return (%)					Sharpe ratio				
	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$
EM = 50%										
DNN-RW	-0.75	0.24	-0.73	-1.53	-3.30	-1.93	0.08	-0.13	-0.16	-0.21
CW-RW	-0.55	-0.44	-1.91	-3.27	-3.18	-1.50	-0.19	-0.35	-0.37	-0.25
CW-DNN	**-0.48**	0.04	-0.24	-0.30	0.05	**-1.28**	0.01	-0.05	-0.04	0.00
AHBS-RW	-0.76	0.79	3.40	3.54	3.77	-1.99	0.28	0.51	0.32	0.21
AHBS-DNN	-0.58	1.22	4.17	4.60	4.39	-1.49	0.43	0.63	0.42	0.25
AHBS-VAR	-0.87	-0.26	-0.25	-1.89	-4.07	-2.33	-0.10	-0.04	-0.17	-0.23
AE-LSTM	-0.76	-0.03	1.67	1.90	0.01	-1.92	-0.02	0.23	0.15	0.00
fRW	-0.56	**1.98**	4.97	4.05	3.34	-1.41	**0.67**	0.74	0.36	0.19
fLinK	-0.65	-0.49	-1.40	-3.44	-3.46	-1.58	-0.17	-0.21	-0.29	-0.20
fGauK	-0.69	0.70	3.38	3.53	3.97	-1.67	0.22	0.55	0.31	0.25
fLapK	-0.76	0.74	5.53	5.93	9.05	-1.86	0.24	0.95	0.58	0.89
fNTK	-0.59	1.26	**6.62**	**8.69**	**13.70**	-1.43	0.42	**1.49**	**1.18**	**1.72**
EM = 75%										
DNN-RW	-1.10	-0.16	-1.12	-1.90	-3.68	-2.81	-0.06	-0.19	-0.20	-0.23
CW-RW	-0.93	-0.85	-2.32	-3.68	-3.60	-2.51	-0.35	-0.43	-0.42	-0.28
CW-DNN	**-0.85**	-0.36	-0.66	-0.73	-0.38	**-2.25**	-0.14	-0.11	-0.08	-0.03
AHBS-RW	-1.15	0.32	2.94	3.11	3.34	-2.99	0.11	0.44	0.28	0.18
AHBS-DNN	-0.95	0.78	3.72	4.19	3.98	-2.40	0.27	0.56	0.38	0.22
AHBS-VAR	-1.25	-0.68	-0.71	-2.35	-4.52	-3.31	-0.25	-0.11	-0.21	-0.26
AE-LSTM	-1.13	-0.47	1.22	1.47	-0.43	-2.81	-0.15	0.17	0.12	-0.03
fRW	-0.98	**1.51**	4.50	3.61	2.92	-2.43	**0.50**	0.66	0.31	0.16
fLinK	-1.06	-0.96	-1.84	-3.90	-3.94	-2.55	-0.33	-0.27	-0.33	-0.22
fGauK	-1.11	0.21	2.93	3.08	3.55	-2.67	0.06	0.47	0.27	0.23
fLapK	-1.18	0.29	5.07	5.50	8.63	-2.85	0.09	0.86	0.53	0.85
fNTK	-1.00	0.78	**6.18**	**8.30**	**13.35**	-2.41	0.25	**1.38**	**1.12**	**1.67**
EM = 100%										
DNN-RW	-1.46	-0.56	-1.53	-2.28	-4.07	-3.66	-0.21	-0.26	-0.23	-0.25
CW-RW	-1.31	-1.25	-2.74	-4.10	-4.03	-3.51	-0.51	-0.50	-0.46	-0.32
CW-DNN	**-1.23**	-0.76	-1.07	-1.17	-0.81	**-3.21**	-0.29	-0.18	-0.12	-0.06
AHBS-RW	-1.55	-0.15	2.46	2.68	2.90	-3.96	-0.06	0.37	0.24	0.16
AHBS-DNN	-1.33	0.33	3.26	3.78	3.56	-3.28	0.11	0.48	0.34	0.20
AHBS-VAR	-1.63	-1.10	-1.18	-2.82	-4.98	-4.26	-0.40	-0.18	-0.25	-0.28
AE-LSTM	-1.50	-0.91	0.78	1.03	-0.87	-3.67	-0.28	0.10	0.08	-0.05
fRW	-1.40	**1.03**	4.02	3.16	2.50	-3.42	**0.34**	0.59	0.27	0.14
fLinK	-1.47	-1.43	-2.29	-4.37	-4.41	-3.50	-0.48	-0.33	-0.36	-0.25
fGauK	-1.54	-0.30	2.48	2.62	3.13	-3.65	-0.10	0.40	0.23	0.20
fLapK	-1.60	-0.17	4.61	5.07	8.20	-3.82	-0.06	0.78	0.49	0.80
fNTK	-1.42	0.30	**5.73**	**7.91**	**12.99**	-3.38	0.09	**1.27**	**1.06**	**1.62**

Table A.14: Mean returns (%) and annualized Sharpe ratio of short delta-neutral straddles with three levels of effective measurement (50%, 75%, and 100%) of all models over the whole test period, from Jan 9, 2019 to Dec 31, 2021. Bold numbers indicate the best-performing model (or models) in a given column.

	Mean return (%)					Sharpe ratio				
	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$
Threshold = 5%										
DNN-RW	0.17	-0.33	-2.26	-4.53	-5.68	0.38	-0.11	-0.33	-0.38	-0.30
CW-RW	0.93	0.98	-2.07	-5.81	-7.48	**2.13**	0.31	-0.29	-0.49	-0.45
CW-DNN	0.76	0.36	-1.83	-2.30	-2.08	1.65	0.10	-0.26	-0.20	-0.14
AHBS-RW	0.96	2.78	7.22	4.57	4.18	1.75	0.78	0.90	0.35	0.20
AHBS-DNN	**1.00**	**3.59**	6.47	5.25	4.27	1.90	**1.05**	0.84	0.41	0.21
AHBS-VAR	0.38	1.14	-0.11	-2.39	-4.76	0.71	0.31	-0.02	-0.18	-0.23
AE-LSTM	-0.35	-0.89	0.30	1.30	-3.85	-0.65	-0.23	0.03	0.09	-0.17
fRW	-0.05	2.20	5.73	4.55	3.81	-0.10	0.58	0.72	0.34	0.18
fLinK	-0.57	-1.25	-1.83	-4.74	-3.80	-0.95	-0.33	-0.22	-0.35	-0.19
fGauK	-0.45	-0.32	5.71	4.59	6.59	-0.76	-0.08	0.78	0.36	0.38
fLapK	-0.35	1.29	6.99	9.63	14.79	-0.60	0.32	1.01	0.83	1.16
fNTK	-0.14	3.56	**10.95**	**15.65**	**20.41**	-0.24	0.90	**1.98**	**1.95**	**2.33**
Threshold = 10%										
DNN-RW	-2.00	-5.47	-8.67	-13.79	-16.02	-3.01	-1.37	-0.95	-0.93	-0.68
CW-RW	**-0.30**	-1.62	-9.03	-12.35	-14.52	**-0.45**	-0.38	-0.97	-0.87	-0.79
CW-DNN	-1.79	-4.28	-9.06	-11.45	-8.57	-2.82	-1.01	-1.02	-0.84	-0.49
AHBS-RW	-2.19	-5.53	-3.69	-2.77	-2.16	-2.86	-1.25	-0.38	-0.19	-0.10
AHBS-DNN	-0.64	**-1.56**	-0.98	-0.93	-1.82	-0.91	**-0.37**	-0.11	-0.07	-0.08
AHBS-VAR	-1.91	-3.83	-7.20	-8.06	-10.93	-2.45	-0.83	-0.72	-0.51	-0.45
AE-LSTM	-2.09	-5.39	-6.13	-7.48	-13.41	-2.66	-1.05	-0.60	-0.44	-0.47
fRW	-5.31	-4.86	-4.76	-3.64	-2.92	-7.22	-1.09	-0.51	-0.24	-0.13
fLinK	-5.80	-5.91	-8.81	-10.90	-10.29	-6.76	-1.26	-0.89	-0.69	-0.45
fGauK	-4.90	-4.57	-3.46	0.79	6.53	-6.16	-0.89	-0.33	0.05	0.32
fLapK	-3.82	-4.64	3.30	9.75	15.29	-4.78	-0.91	0.37	0.80	1.16
fNTK	-5.02	-3.16	**9.97**	**17.34**	**23.10**	-6.30	-0.60	**1.50**	**1.84**	**2.20**

Table A.15: Mean simple returns (%) and annualized Sharpe ratio of all models for short delta-neutral straddles, using two different filtering thresholds of 5% and 10% over the whole test period (Jan 09, 2019, to Dec 31, 2021). Bold numbers indicate the best-performing model (or models) in a given column.

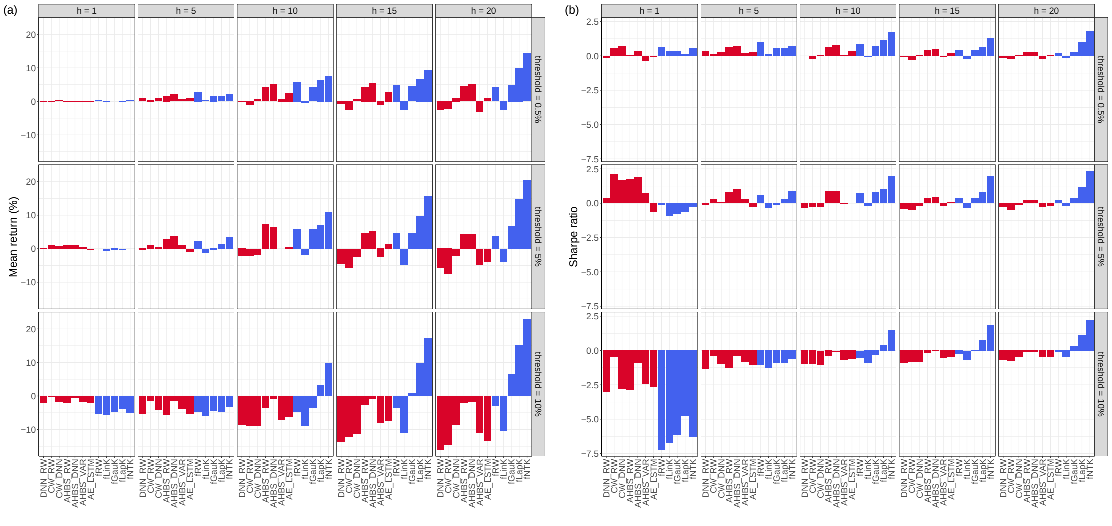


Figure A.10: Mean simple returns in percentage and Sharpe ratio of short delta-neutral straddle strategy using three levels of filtering thresholds: 0.5%, 5% and 10%. The prediction period is from Jan 09, 2019 to Dec 31, 2021.

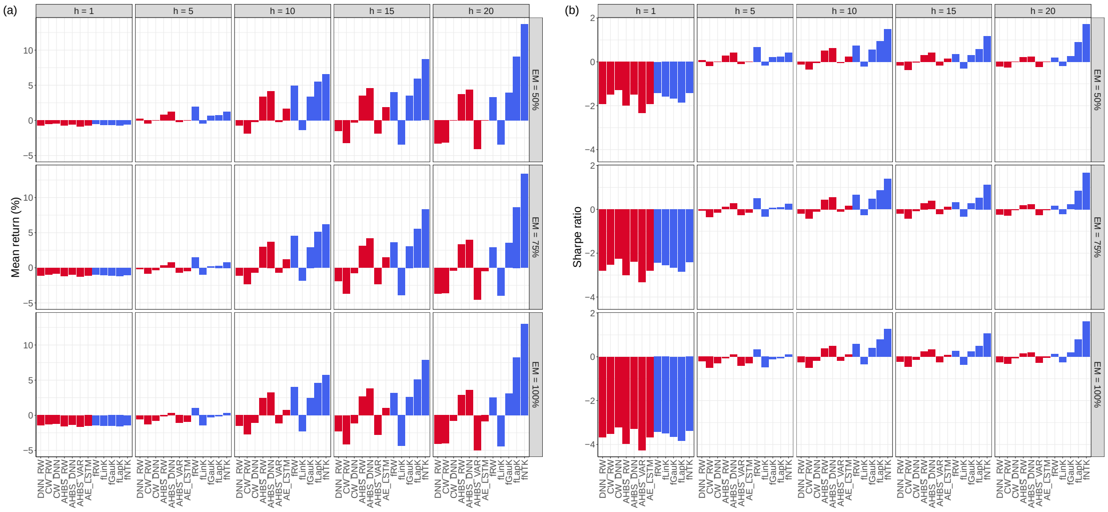


Figure A.11: Mean simple returns in percentage and Sharpe ratio of short delta-neutral straddle strategy using three levels of effective measure (EM): 50%, 75% and 100%. The prediction period is from Jan 09, 2019 to Dec 31, 2021.

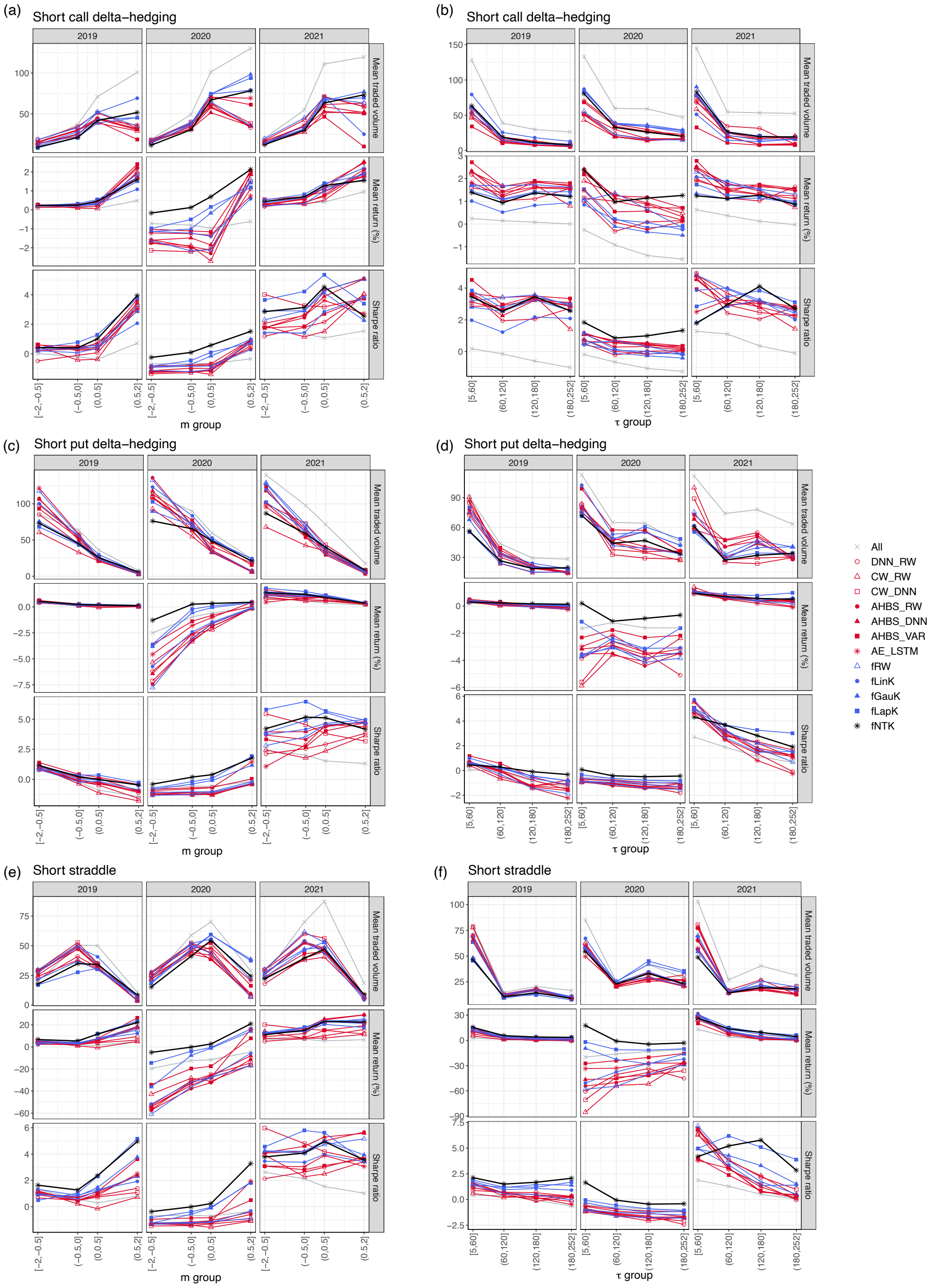


Figure A.12: Mean traded volume, mean simple returns in percentage, and annualized Sharpe ratio of the models for short call delta-hedging, short-put delta-hedging, and short delta-neutral straddle at the forecasting horizon $h = 20$, across different moneyness m groups $[-2,-0.5]$, $(-0.5,0]$, $(0,0.5]$, and $(0.5,2]$, and time-to-maturity τ groups $[5,60]$, $(60,120]$, $(120,180]$, and $(180,252]$, and in three prediction periods: 2019, 2020, and 2021.

	Mean return (%)					Sharpe ratio				
	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$	$h=1$	$h=5$	$h=10$	$h=15$	$h=20$
Short delta-hedging										
DNN-RW	-0.21	-0.41	-2.04	-4.38	-6.35	-0.55	-0.17	-0.36	-0.40	-0.37
CW-RW	-0.43	-1.48	-6.22	-8.36	-8.78	-0.97	-0.27	-0.98	-0.74	-0.56
CW-DNN	-0.10	-0.34	-1.88	-4.54	-4.39	-0.28	-0.13	-0.31	-0.41	-0.29
AHBS-RW	-0.32	-0.89	-3.07	-5.97	-8.41	-0.82	-0.34	-0.51	-0.53	-0.46
AHBS-DNN	-0.27	-0.42	-2.46	-4.94	-7.34	-0.63	-0.16	-0.41	-0.44	-0.42
AHBS-VAR	-0.55	-0.70	-4.50	-7.26	-10.15	-1.37	-0.23	-0.66	-0.57	-0.52
AE-LSTM	-0.09	-0.31	0.09	-3.51	-13.70	0.17	0.07	0.01	-0.27	-0.57
fRW	**0.33**	1.81	-0.74	-3.76	-7.52	**0.74**	0.53	-0.11	-0.30	-0.32
fLinK	0.06	-0.80	-5.01	-7.53	-9.94	0.09	-0.26	-0.69	-0.57	-0.50
fGauK	0.10	-0.29	0.81	0.73	6.10	0.17	-0.08	0.14	0.07	0.67
fLapK	-0.15	0.20	3.66	5.32	7.03	-0.30	0.06	0.68	0.72	0.78
fNTK	-0.13	**3.37**	**5.51**	**7.99**	**9.35**	-0.29	**0.78**	**1.25**	**1.75**	**0.94**
Simple short straddle										
DNN-RW	-0.16	-0.31	-2.08	-3.90	-5.40	-0.46	-0.14	-0.37	-0.40	-0.34
CW-RW	-0.48	-2.76	-6.21	-7.93	-8.07	-1.25	-1.07	-1.00	-0.76	-0.58
CW-DNN	-0.11	-0.22	-1.90	-4.28	-3.67	-0.32	-0.09	-0.32	-0.42	-0.27
AHBS-RW	-0.29	-0.79	-2.97	-5.55	-7.57	-0.79	-0.34	-0.51	-0.54	-0.45
AHBS-DNN	-0.12	-0.40	-2.47	-4.72	-6.58	-0.35	-0.18	-0.42	-0.46	-0.41
AHBS-VAR	-0.42	-0.91	-4.35	-6.77	-9.44	-1.19	-0.36	-0.66	-0.58	-0.53
AE-LSTM	-0.03	-0.28	0.33	-2.73	-12.20	-0.08	-0.10	0.04	-0.23	-0.58
fRW	**0.19**	1.24	-0.46	-3.78	-5.09	**0.47**	0.45	-0.08	-0.33	-0.29
fLinK	-0.32	-0.61	-4.82	-7.59	-9.14	-0.79	-0.22	-0.68	-0.64	-0.50
fGauK	-0.27	-0.58	1.31	1.51	6.41	-0.67	-0.20	0.24	0.17	0.74
fLapK	-0.34	0.46	3.88	5.23	7.48	-0.84	0.15	0.78	0.73	0.88
fNTK	-0.24	**2.13**	**5.52**	**7.95**	**10.10**	-0.59	**0.77**	**1.36**	**1.81**	**1.24**
Delta-neutral short straddle										
DNN-RW	-0.15	-0.32	-2.09	-3.93	-5.47	-0.44	-0.15	-0.37	-0.40	-0.34
CW-RW	-0.49	-2.79	-6.23	-8.01	-8.17	-1.27	-1.08	-1.01	-0.77	-0.58
CW-DNN	-0.11	-0.25	-1.92	-4.34	-3.77	-0.33	-0.10	-0.32	-0.42	-0.28
AHBS-RW	-0.29	-0.79	-2.97	-5.57	-7.63	-0.79	-0.34	-0.51	-0.54	-0.45
AHBS-DNN	-0.12	-0.40	-2.46	-4.73	-6.62	-0.35	-0.18	-0.42	-0.46	-0.41
AHBS-VAR	-0.42	-0.93	-4.36	-6.80	-9.52	-1.19	-0.36	-0.66	-0.59	-0.53
AE-LSTM	-0.03	-0.31	0.28	-2.84	-12.35	-0.07	-0.11	0.04	-0.24	-0.59
fRW	**0.19**	1.21	-0.50	-3.85	-5.22	**0.47**	0.44	-0.08	-0.34	-0.29
fLinK	-0.32	-0.65	-4.84	-7.68	-9.20	-0.79	-0.24	-0.69	-0.64	-0.50
fGauK	-0.28	-0.62	1.27	1.42	6.37	-0.69	-0.21	0.23	0.15	0.73
fLapK	-0.35	0.42	3.83	5.18	7.41	-0.87	0.14	0.77	0.72	0.87
fNTK	-0.24	**2.08**	**5.49**	**7.93**	**10.03**	-0.60	**0.75**	**1.35**	**1.80**	**1.21**

Table A.16: Mean simple return (%) and annualized Sharpe ratio of short delta-hedging (using both call and put options), simple short straddle (consisting of one call and one put option), and short delta-neutral straddle for at-the-money options, defined by $|\Delta| \in [0.48, 5.02]$, where Δ is the delta of the options. Bold numbers indicate the best-performing model (or models) in a given column.

References

Almeida, C., J. Fan, G. Freire, and F. Tang (2022). “Can a Machine Correct Option Pricing Models?” *Journal of Business & Economic Statistics*, 41.3, 995–1009.

Carr, P. and L. Wu (2016). “Analyzing volatility risk and risk premium in option contracts: A new theory”. *Journal of Financial Economics*, 120.1, 1–20.

Chen, X. (2007). “Large sample sieve estimation of semi-nonparametric models”. *Handbook of Econometrics*, 6, 5549–5632.

Chen, Y. and B. Li (2017). “An adaptive functional autoregressive forecast model to predict electricity price curves”. *Journal of Business & Economic Statistics*, 35.3, 371–388.

Corsi, F. (2009). “A simple approximate long-memory model of realized volatility”. *Journal of Financial Econometrics*, 7.2, 174–196.

Daniely, A., R. Frostig, and Y. Singer (2016). “Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity”. *Advances in Neural Information Processing Systems*, 29.

Diebold, F. X. and R. S. Mariano (2002). “Comparing predictive accuracy”. *Journal of Business & Economic Statistics*, 20.1, 134–144.

Dumas, B., J. Fleming, and R. E. Whaley (1998). “Implied volatility functions: Empirical tests”. *The Journal of Finance*, 53.6, 2059–2106.

Goncalves, S. and M. Guidolin (2006). “Predictable dynamics in the S&P 500 index options implied volatility surface”. *The Journal of Business*, 79.3, 1591–1635.

Goyal, A. and A. Saretto (2009). “Cross-section of option returns and volatility”. *Journal of Financial Economics*, 94.2, 310–326.

Gu, S., B. Kelly, and D. Xiu (2021). “Autoencoder asset pricing models”. *Journal of Econometrics*, 222.1, 429–450.

Hochreiter, S. and J. Schmidhuber (1997). “Long short-term memory”. *Neural Computation*, 9.8, 1735–1780.

Hsing, T. and R. Eubank (2015). *Theoretical foundations of functional data analysis, with an introduction to linear operators*. Vol. 997. John Wiley & Sons.

Jacot, A., F. Gabriel, and C. Hongler (2018). “Neural tangent kernel: Convergence and generalization in neural networks”. *Advances in Neural Information Processing Systems*, 31.

Klepsch, J. and C. Klüppelberg (2017). “An innovations algorithm for the prediction of functional linear processes”. *Journal of Multivariate Analysis*, 155, 252–271.

Ledoit, O. and M. Wolf (2008). “Robust performance hypothesis testing with the Sharpe ratio”. *Journal of Empirical Finance*, 15.5, 850–859.

Morris, J. S., M. Vannucci, P. J. Brown, and R. J. Carroll (2003). “Wavelet-based nonparametric modeling of hierarchical functions in colon carcinogenesis”. *Journal of the American Statistical Association*, 98.463, 573–583.

Ramsay, J. and B. Silverman (2005). *Functional Data Analysis*. Springer Series in Statistics. Springer. ISBN: 9780387400808.

Schölkopf, B., R. Herbrich, and A. J. Smola (2001). “A generalized representer theorem”. *International Conference on Computational Learning Theory*. Springer, 416–426.

Yao, F., H.-G. Müller, and J.-L. Wang (2005). “Functional Data Analysis for Sparse Longitudinal Data”. *Journal of the American Statistical Association*, 100.470, 577–590.

Zhang, W., L. Li, and G. Zhang (2023). “A two-step framework for arbitrage-free prediction of the implied volatility surface”. *Quantitative Finance*, 23.1, 21–34.