Forecasting day-ahead EURUSD tail risk: Leveraging machine learning and the volatility surface.

Morten Risstad[a,*,1], Ida. A. Moen[a], Marie. S. Pedersen[a], Hans Magnus Utne[a] and Sjur Westgaard[a]

[a]*Norwegian University of Science and Technology, Dep. of Industrial Economics and Technology Management, Alfred Getz v. 4, 7004, Trondheim, Norway*

ARTICLE INFO

Keywords:
Finance
Expected shortfall
Implied volatility surface
Variance risk premia
Foreign exchange
Forecasting
Machine learning

ABSTRACT

This paper estimates expected shortfall for the EURUSD exchange rate. We contribute to the literature along two important dimensions. First, we expand the set of predictors to ~~include~~ not only at-the-money implied volatilities, which is the predominant approach in the literature, ~~to~~ but also include variance risk premia and risk reversals. The former corrects for known forecasting bias in implied volatilities. The latter, which is directly observable in option markets, contains predictive information about the shape of the conditional return distribution. Our second contribution is to analyze the performance of a broad set of supervised machine learning models, including tree-based models and ~~recurring~~ recurrent neural networks. This class of models is particularly appealing in our context, as they have inherent capabilities to deal with high-dimensional sets of potentially correlated predictors with time-varying non-linear dynamics. We find that machine learning models generally benefit from increasing the set of predictors and outperform relevant benchmark models. This is particularly pronounced when the euro depreciates. Among the tested models, CatBoost emerges as the top performer. We verify that these results hold under alternative loss functions, across multiple model specifications and combinations, and are stable over time. Furthermore, machine learning models tend to outperform more profoundly during periods of higher market uncertainty, which is particularly appealing.

1. Introduction

Investment decisions expose investors to uncertain future payoffs. Appropriate models for tail risk play an important part in assessing conditional return distributions. Since its introduction in the mid 1990s Value-at-Risk (VaR) has been a popular risk measure among both practitioners and regulators (Basel Committee on Banking Supervision, 1996). More recently, regulators have endorsed Expected Shortfall (ES), which addresses limitations of VaR (Basel Committee on Banking Supervision, 2016). VaR and ES are currently widely applied for quantitative risk measurement.

This study is concerned with predicting and backtesting the expected shortfall of the next-day EURUSD spot exchange rate. For this purpose, we use predictors derived from the implied volatility (IV) surface. The over-the-counter (OTC) foreign exchange (FX) options market is organized in a manner which is particularly well suited for extracting predictive information. In the interbank OTC FX options market, market makers quote and trade a set of new options every day, with standardized maturities and moneyness.[1] This is in contrast to exchange-traded options

*Corresponding author
morten.risstad@ntnu.no (M. Risstad)
ORCID(s): 0000-0003-2562-8892 (M. Risstad); 0000-0003-4891-2730 (S. Westgaard)

[1]Standardized maturities range from one day through 1:2:3 weeks, 1:2:3:4:5:6:9:12 months and up to two years. Moneyness is standardized by quoting strikes in terms of option deltas, ranging from 50 delta (at-the-money) to far out-of-the-money puts and calls (i.e. 5 delta).

for which new maturities are introduced once a month or even less frequently. In OTC FX options, we therefore have consistent time series to investigate - for example, one-week implied volatilities over time - which leaves less margin for errors in the estimation of time series properties. In addition, the OTC FX options market has the beneficial property of quoting market prices in terms of implied volatilities directly. Thus, implied volatilities need not be inverted from option premia. This is possible only since the OTC FX market, unlike any other market, has standardized the option formula used for quoting implied volatilities.[2] These institutionalized features of OTC FX options interbank trading make this market especially convenient for empirical analysis.[3]

IV represents volatility expectations of representative risk-neutral agents assuming a log-normal model for the underlying security. This inherent forward-looking nature makes implied volatilities appealing for forecasting purposes. The predictive power of implied volatility has been explored across asset classes, see for instance the recent review by Gunnarsson, Isern, Kaloudis, Risstad, Vigdel and Westgaard (2024) and references therein. The consensus is that IV is helpful, also when augmented in realized volatility (RV) models relying on high-frequency data. Regarding foreign exchange markets more specifically, the literature suggest that IV is beneficial and in many cases dominate RV (Xu and Taylor, 1995; Busch, Christensen and Nielsen, 2011; Plíhal and Lyócsa, 2021; Lyócsa, Plíhal and Vỳrost, 2024). Still, in the presence of risk-averse agents, IV is a biased predicted of the true future volatility, due to the existence of risk premia. Andersen, Fusari and Todorov (2020) finds that this variance risk premium (VRP) is associated with adverse tail events and displays persistent shifts unrelated to volatility. Hence, even though VRP potentially acts as a correction for bias in IV, it also creates time-varying estimation errors, which presumably are higher in periods of increased risk aversion.

The typical approach in the literature is to rely on at-the-money options as proxies for IV. This is reasonable, since these options have the highest liquidity and hence less measurement errors. Still, one could argue that out-of-the money options are more informative about tail events. A few examples exist, where attempts to incorporate information from the implied volatility skewness are made. Christoffersen and Mazzotta (2005) find that option combinations improve FX density forecasts. de Lange, Risstad and Westgaard (2022) find that risk reversals contain predictive information for EURUSD VaR using a linear quantile regression model and Blom, de Lange and Risstad (2023) report similar results using supervised machine learning models. Using a similar dataset as this paper, albeit a fixed estimation window, Risstad, Westgaard, Moen, Pedersen and Utne (2024) find that machine learning models, in contrast to the traditional econometric models, benefit from increasing the dimensions of the feature space when estimating VaR.

Haug, Frydenberg and Westgaard (2010) discuss the statistical properties of IV in major currencies, and point out that supply-and-demand–based pricing (Garleanu, Pedersen and Poteshman, 2008) also applies to OTC FX option markets. For example, if liquidity is reduced for a certain strike and maturity, and this option is in demand, this could

[2] The standard is what is known as the Garman-Kolhagen (Garman and Kohlhagen, 1983) version of the Black-Scholes-Merton model.
[3] See Reiswich and Wystup (2010) for an introduction to the OTC FX option market and related quoting conventions.

severely impact the option price and the IV. Arguably, supply-demand preferences vary between clusters of market participants. Hedge funds will attempt to exploit perceived mispricings across the full volatility surface, while corporate hedgers often have predefined hedging programs in place and might be less sensitive to short-term fluctuations in IVs. Financial intermediaries and market-makers, on the other hand, typically have limited capital and are thus incentivized to reduce overall market risk. Haug and Taleb (2011) argue that, although delta-hedging is frequently applied, traders generally prefer to hedge option positions by entering into other option contracts with similar strikes and maturities. This relates to the non-linear exposure of options to extreme market changes. More specifically, and omitting technical details, this behaviour is motivated by vega (sensitivity to changes in implied volatility) and gamma (sensitivity to jumps in the underlying exchange rate) risk of options. As outlined in Haug et al. (2010) market makers typically manage vega and gamma risk by trading short-dated options. Hence, a reasonable conjecture is that the IV surface contains predictive information, not only about the overall level of future realized volatility, but also about the likelihood and magnitude of extreme price movements.

To this end, our empirical approach resembles that of Lyócsa et al. (2024). In line with Lyócsa et al. (2024) we use IVs from options with daily, weekly, and monthly expiry, motivated by the heterogenous market participant hypothesis. Furthermore, we estimate traditional econometric models, specifically the EGARCH-model of Nelson (1991) and the linear quantile regression model of Koenker and Bassett (1978), as well as the joint VaR and ES framework proposed by Dimitriadis and Bayer (2019). Additionally, we estimate a broad set of supervised learning models, which distinguishes our empirical analysis from that of Lyócsa et al. (2024). Even though machine learning models have been shown to be accurate for VaR forecasts (Keilbar and Wang, 2022; Chronopoulos, Raftapostolos and Kapetanios, 2024), it is not clear whether this equally applies to ES forecasts.

This paper contributes to the literature in two important dimensions. First we investigate the relevance of RR and VRP as predictors of ES, when combined with IV. Second, as far as we know, this paper is the first to investigate the performance of tree-based machine learning models and neural networks for this purpose.

2. Data and variables

Our dataset spans nearly 17 years, from January 2007 to December 2022. Daily observations of the EURUSD spot exchange rate and implied volatility surface are sourced from Bloomberg. Bloomberg aggregates quotes across brokers, large banks, and insurance companies. This reduces idiosyncratic market microstructure noise due to circumstances related to specific market participants.

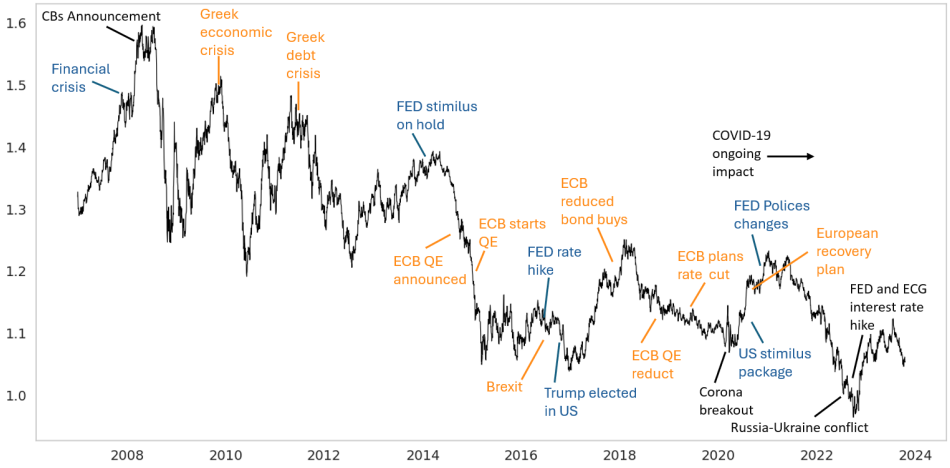

Figure 1: The EURUSD rate and major economic events, adapted from Lyócsa et al. (2024).

As predictors, we use implied volatility (IV), risk-reversals (RR) and variance risk premia (VRP) with daily, weekly, and monthly time to expiry. At-the-money mid-quotes represent IV. The risk reversal $RR_{\Delta,t,T}$ is defined as

$$RR_{\Delta,t,T} = \sigma^{call}_{\Delta,t,T} - \sigma^{put}_{\Delta,t,T} \tag{1}$$

The risk reversal is the an option combination whose price at time t is determined by the difference in IVs of an out-of-the-money foreign currency call option $\sigma^{call}_{\Delta,t,T}$ and an out-of-the-money put option $\sigma^{put}_{\Delta,t,T}$, both with delta equal to Δ and time-to-expiry equal to T. The price of the RR is zero when the risk-neutral distribution or the underlying spot rate is symmetric, meaning that the value of being long the call exactly offsets the value of being short the put. Moreover, if the distribution is negatively (positively) skewed, the price of the risk-reversal is negative (positive), which further implies that the market expects the currency to depreciate (appreciate). Hence, the risk-reversal can be used to assess how the market sees the balance of risks between a large appreciation and a large depreciation in the exchange rate. A large positive reading of EURUSD risk-reversals suggests that higher probabilities are attached to a sizeable appreciation of the euro, whereas a large negative reading indicates expectations skewed in favor of a considerable euro depreciation. In this paper, we use risk-reversals with $\Delta = 0.25$.

As IV can be interpreted as the risk-neutral expectation of volatility, it is by definition a biased predictor of future physical volatility in the presence of risk-aversion. Following Bollerslev, Tauchen and Zhou (2009), Bekaert and Hoerova (2014), Slim, Dahmene and Boughrara (2020), among others, we estimate the variance risk premium by

Table 1
Descriptive statistics.

	Spot returns	IV^D	IV^W	IV^M	RR^D	RR^W	RR^M	VRP^D	VRP^W	VRP^M
Mean	-0.004	9.061	9.066	9.072	-0.276	-0.333	-0.592	$-8.000 \cdot 10^{-5}$	$-8.307 \cdot 10^{-6}$	$-9.154 \cdot 10^{-6}$
Std	0.573	3.922	3.775	3.477	0.529	0.622	0.818	$2.600 \cdot 10^{-5}$	$2.333 \cdot 10^{-5}$	$2.549 \cdot 10^{-5}$
Skewness	0.121	1.476	1.590	1.418	-1.312	-1.103	-1.159	-5.213	-7.146	-6.981
Kurtosis	2.576	3.536	4.138	3.304	3.993	4.275	1.894	75.279	109.872	93.965
Max	3.800	33.887	33.584	28.882	2.740	2.948	2.895	$1.540 \cdot 10^{-4}$	$9.691 \cdot 10^{-5}$	$8.343 \cdot 10^{-5}$
Min	-2.741	1.775	2.737	3.772	-3.905	-4.005	-4.193	$-5.580 \cdot 10^{-4}$	$-5.345 \cdot 10^{-4}$	$-5.532 \cdot 10^{-4}$
ρ_1	0.004	0.760	0.975	0.993	0.888	0.951	0.990	0.074	0.255	0.369

Superscript "D", "W" or "M" denote daily, weekly and monthly time to expiry. ρ_1 is the first-order autocorrelation coefficient.

Table 2
Correlation matrix.

	Spot returns	IV^D	IV^W	IV^M	RR^D	RR^W	RR^M	VRP^D	VRP^W	VRP^M
Spot returns	1.00									
IV^D	0.00	1.00								
IV^W	-0.02	0.87	1.00							
IV^M	-0.01	0.83	0.95	1.00						
RR^D	-0.03	-0.33	-0.33	-0.32	1.00					
RR^W	-0.02	-0.27	-0.34	-0.31	0.90	1.00				
RR^M	-0.01	-0.31	-0.38	-0.40	0.82	0.84	1.00			
VRP^D	0.01	-0.01	-0.03	-0.03	-0.05	-0.05	-0.08	1.00		
VRP^W	0.02	0.02	0.01	0.02	-0.05	-0.05	-0.08	0.84	1.00	
VRP^M	0.03	0.04	0.04	0.05	-0.07	-0.07	-0.10	0.83	0.95	1.00

Pearson's linear correlation coefficients. Superscript "D", "W" or "M" denote daily, weekly and monthly time to expiry.

subtracting the expected realized volatility from the squared implied volatility;

$$VRP_{t,T} = E_t^Q(V_{t,T}^2) - E_t^P(V_{t,T}^2) \tag{2}$$

Here, $V_{t,T}^2$ denotes return variation, $E_t^Q(V_{t,T}^2)$ represents the preliminary of the variance under the risk-neutral probability, and $E_t^P(V_{t,T}^2)$ denotes the expected variance prediction under the physical probability measure, proxied by ex-post realized variance. To estimate $E_t^P(V_{t,T}^2)$ we resample intraday tick-level data from Dukas Copy[4] to 5-minute intervals as recommended by Liu, Patton and Sheppard (2015), following the procedure proposed by Barndorff-Nielsen, Hansen, Lunde and Shephard (2008).[5]

[4] publicly available at https://www.dukascopy.com/swiss/english/marketwatch/historical/

[5] First, we delete entries with (i) zero quotes, (ii) negative bid-ask spread, (iii) bid-ask spread greater than 50 times the median spread on that day and (iv) for which the mid-quote deviates by more than ten mean absolute deviations from a centered mean, excluding the observation under consideration) of 25 observations before and 25 observations after. Second, we compute mid-quotes as the average of bid and ask quotes and resample the data using a 5-min frequency.

On day t, the consistent estimator of the true latent variance is

$$RV_t^2 \equiv \sum_{i=1}^{M} r_{t,i}^2, \tag{3}$$

where $M = 1/\Delta$, and the Δ-period intraday return is $r_{t,i} \equiv \log(P_{t-1+i\times\Delta}) - \log(P_{t-1+(i-1)\times\Delta})$, where P is the EURUSD spot exchange rate.

~~Standard diagnostic tests reveal that the variables in Table [redacted] are stationary, heteroscedastic, and non-normal. Furthermore, variance inflation factor analysis indicate multicollinearity, as expected.~~

Dickey-Fuller, Breusch-Pagan, and Breusch-Godfrey tests at 5% confidence level confirm that the variables in Table 1 are stationary, heteroscedastic, and non-normally distributed. The correlation matrix in Table 2 shows high and positive correlations within each block of predictors (IV, RR, and VRP). Intrablock correlations greater than 0.8 indicate that predictors of daily, weekly and monthly time-to-maturity are driven by common latent factors. Moreover, IVs are moderately negatively correlated with RRs, whereas VRPs are uncorrelated with other predictors. Correlations between blocks of independent variables relatively close to zero indicate that IV, RR, and VRP do not share common predictive information.

3. Methodology

3.1. VaR and ES as measures of market risk

VaR is a widely recognized statistical measure used to quantify market risk over a target horizon (Jorion, 1996). VaR at confidence level α is defined as

$$\text{VaR}_\alpha = \min\{m : P(L \leq m) \geq 1 - \alpha\}, \tag{4}$$

where VaR_α is the smallest monetary amount m such that the probability of the loss L being less than or equal to m is at least 1-α. Since its introduction in the 1980s, VaR has become a widely adopted measure of financial risk. Despite widely adopted in the financial industry, VaR has some inherent, significant weaknesses. One critical weakness of VaR is its inability to account for the severity of losses beyond its predefined threshold. This oversight ~~might~~ may lead to underestimation of tail risk, as portfolios ~~sharing~~ with the same VaR_α can incur different losses beyond the α threshold. Furthermore, VaR does not qualify as a coherent risk measure due to its lack of subadditivity (Acerbi and Tasche, 2002). ~~Expected Shortfall (ES), also known as Conditional VaR or Tail VaR, is a coherent and more robust risk measure through estimating the average of the most severe losses, conditional upon those losses exceeding the VaR threshold~~ Expected Shortfall (ES), also referred to as Conditional Value-at-Risk or Tail Value-at-Risk, constitutes a coherent and more comprehensive risk measure, as it evaluates the conditional expectation of portfolio losses given that they exceed the corresponding VaR level (Taylor, 2019):

$$ES_\alpha = \frac{1}{\alpha}\int_0^\alpha VaR_u du \tag{5}$$

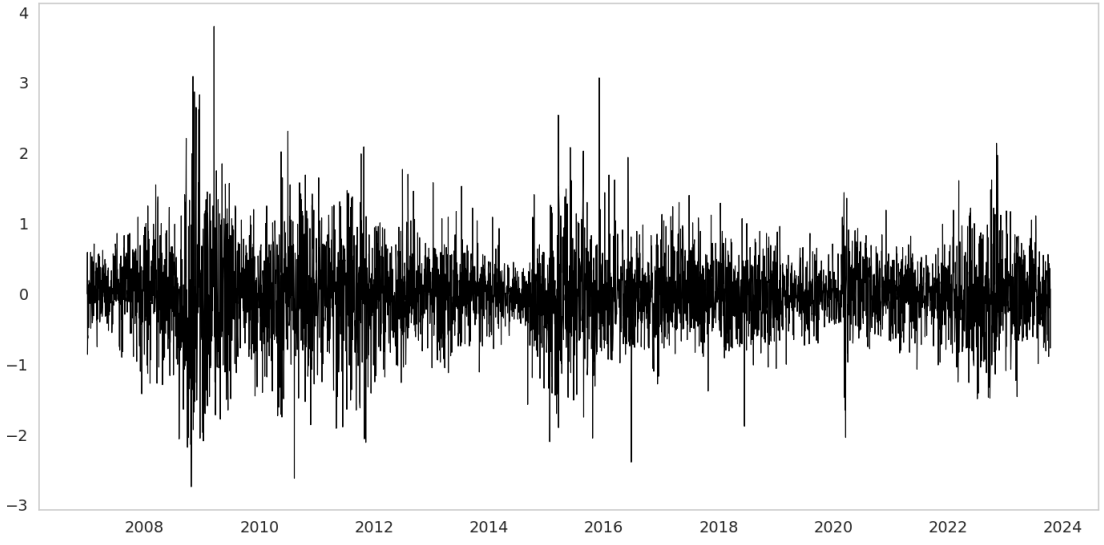

Figure 2: EURUSD spot returns

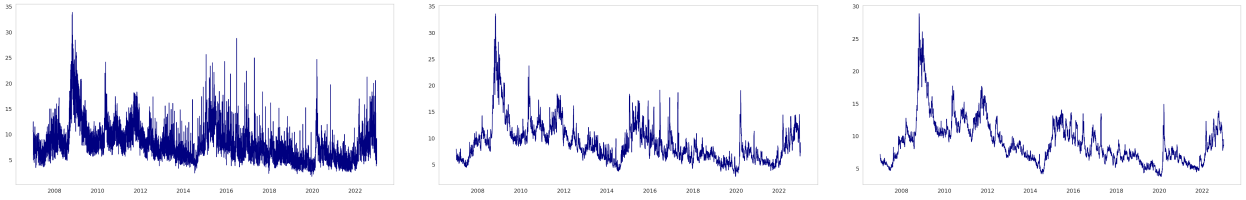

Figure 3: Implied volatility

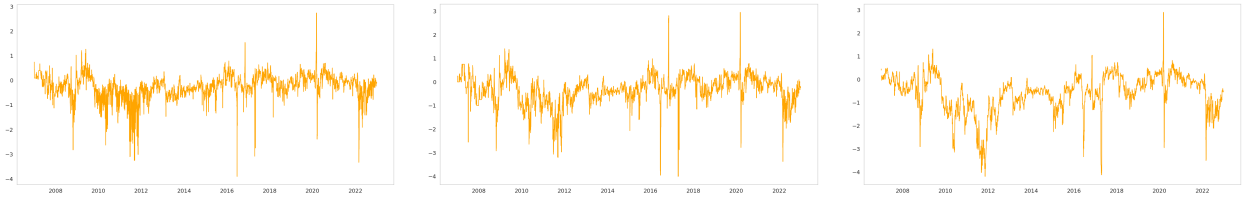

Figure 4: Risk reversal

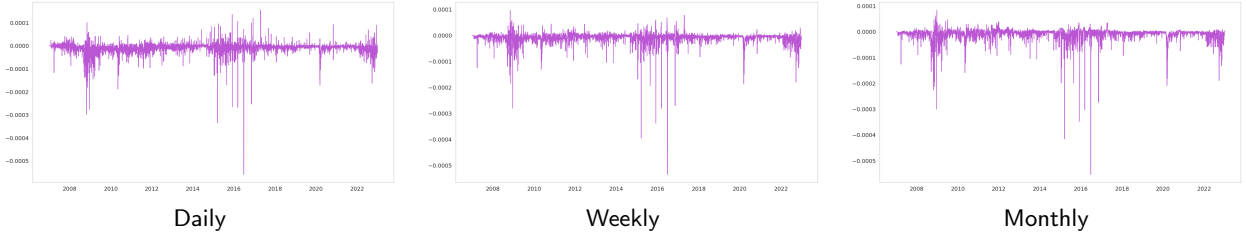

Figure 5: Variance risk premia

Despite the theoretical appeal of ES, its practical application is inherently challenging. ES is not elicitable, which means no scoring function exists whose expected value is minimized only when the forecast aligns with the true ES (Gneiting, 2011). Furthermore, since the true tail density of the conditional tail distribution is generally unknown, applying parametric models is challenging. To approximate the integral in Equation 5, we follow Lyócsa et al. (2024) and use the following expression:

$$ES_{R_t}(\alpha) \approx \frac{1}{p}\sum_{j=1}^{p}\widehat{VaR}_{R_t}(u_j), \tag{6}$$

where $ES_{R_t}(\alpha)$ is ES at confidence level α and $VaR_{R_t}(u_j)$ is VaR at confidence level u_j, $j \in \{1, 2, \dots, p\}$. Eq. (6) approximates ES at confidence level α as a weighted sum of p VaR estimates for quantiles beyond α. While increasing p increases the precision of ES approximations, this simultaneously increases complexity. To balance accuracy and computational costs, we follow Lyócsa et al. (2024) and set $p = 5$.[6]

3.2. Models

As outlined in Section 2 we employ three explanatory variables (IV, RR, and VRP) spanning three different times to expiry (daily, weekly, and monthly). This leads us to four different specifications for each of the model classes investigated.

3.2.1. Quantile regression

Introduced by Koenker and Bassett (1978), quantile regression models quantiles of the conditional distribution of the response variable from a set of observed covariates. Due to its relevance for tail risk quantifications, quantile regressions have found a large number of applications across asset classes, including the EURUSD exchange rate, see Haugom, Ray, Ullrich, Veka and Westgaard (2016); Lyócsa, Plíhal and Vỳrost (2021); de Lange et al. (2022); Lyócsa et al. (2024) among many others.

The four quantile regression models include[7];

the IV model:

$$\widehat{VaR}_{R_{t+1}}(u_j) = \hat{\beta}_{0,u_j} + \hat{\beta}_{1,u_j} IV_t^D + \hat{\beta}_{2,u_j} IV_t^W + \hat{\beta}_{3,u_j} IV_t^M + \epsilon_{u_j,t} \tag{7}$$

[6]Setting $p = 5$ in (6) divides the tail of the conditional return distribution into five uniformly distributed quantiles. For instance, we approximate the 5% ES by the weighted sum of 5%, 4%, 3%, 2% and 1% VaR estimates. Similarly, the 97.5% ES is a function of 97.5%, 98.0%, 98.5%, 99.0, and 99.5% VaRs.%

[7]Similarly, we estimate comparable specifications for the remaining models.

the IV & RR model:

$$\widehat{VaR}_{R_{t+1}}(u_j) = \hat{\beta}_{0,u_j} + \hat{\beta}_{1,u_j} IV_t^D + \hat{\beta}_{2,u_j} IV_t^W + \hat{\beta}_{3,u_j} IV_t^M + \hat{\beta}_{4,u_j} RR_t^D + \hat{\beta}_{5,u_j} RR_t^W + \hat{\beta}_{6,u_j} RR_t^M + \epsilon_{u_j,t} \tag{8}$$

the IV & VRP model:

$$\widehat{VaR}_{R_{t+1}}(u_j) = \hat{\beta}_{0,u_j} + \hat{\beta}_{1,u_j} IV_t^D + \hat{\beta}_{2,u_j} IV_t^W + \hat{\beta}_{3,u_j} IV_t^M + \hat{\beta}_{4,u_j} VRP_t^D + \hat{\beta}_{5,u_j} VRP_t^W + \hat{\beta}_{6,u_j} VRP_t^M + \epsilon_{u_j,t} \tag{9}$$

the IV, RR & VRP model:

$$\begin{aligned}\widehat{VaR}_{R_{t+1}}(u_j) = {} & \hat{\beta}_{0,u_j} + \hat{\beta}_{1,u_j} IV_t^D + \hat{\beta}_{2,u_j} IV_t^W + \hat{\beta}_{3,u_j} IV_t^M + \\ & \hat{\beta}_{4,u_j} RR_t^D + \hat{\beta}_{5,u_j} RR_t^W + \hat{\beta}_{6,u_j} RR_t^M + \\ & \hat{\beta}_{7,u_j} VRP_t^D + \hat{\beta}_{8,u_j} VRP_t^W + \hat{\beta}_{9,u_j} VRP_t^M + \epsilon_{u_j,t},\end{aligned} \tag{10}$$

where $\widehat{VaR}_{R_{t+1}}(u_j)$ is the estimated VaR at quantile level u_j for period $t+1$ and $\hat{\beta}_{n,u_j}$ are coefficients with $n = 0, 1, \ldots, p$ associated parameters.

3.2.2. GARCH

The Exponential Generalized Autoregressive Conditional Heteroskedasticity (EGARCH) model (Nelson, 1991) explicitly accounts for the asymmetric impacts of shocks on volatility. ~~The Exponential Generalized Autoregressive Conditional Heteroskedasticity (EGARCH) model~~ (Nelson, 1991) ~~explicitly accounts for the asymmetric impacts of shocks on volatility.~~ The general structure of the GARCH model (Bollerslev, 1987) is

$$R_t = \mu_0 + \phi R_{t-1} + \epsilon_t, \tag{11}$$

$$\epsilon_t = \sigma_t \eta_t, \tag{12}$$

$$\eta_t \sim \text{i.i.d. } (0, 1), \tag{13}$$

where R_t is the return at time t, μ_0 is the mean return, ϕ is the autoregressive coefficient, and ϵ_t represents the error term with σ_t indicating the time-varying standard deviation. The η_t term denotes standardized i.i.d innovations, for which we employ the skewed Student's t-distribution.

The EGARCH model is specified as:

$$\log \sigma_t^2 = \omega + \alpha z_{t-1} + \gamma(|\eta_{t-1}| - \mathbb{E}|\eta_{t-1}|) + \beta \log \sigma_{t-1}^2 + \sum_{k=1}^{K} \nu_k X_{k,t-1}, \tag{14}$$

where z_{t-1} represents the lagged standardized residual. The parameters α, γ and β estimate the impacts of the lagged squared residual, the lagged shock, and the lagged variance, respectively. Finally, ν_k and $X_{k,t-1}$ are coefficients and values of exogenous regressors.

3.2.3. Joint VaR and ES regression framework

The joint VaR and ES model of Dimitriadis and Bayer (2019), henceforward *DB*, establishes a framework for direct, joint estimation of VaR and ES using regressions and a choice-based loss function. It incorporates returns $R \in \mathbb{R}$ and explanatory variables $X_q \in \mathbb{R}^{k_1}$ and $X_e \in \mathbb{R}^{k_2}$, $k_1, k_2 \in \mathbb{N}$. At a given confidence level $\tau \in (0,1)$, the regression is formalized by the equations:

$$R = X_q' \theta_0^q + u^q \quad \text{and} \quad R = X_e' \theta_0^e + u^e \tag{15}$$

where θ_0^q and θ_0^e are the parameter vectors for VaR and ES, respectively, and u^q and u^e represent the corresponding error terms. These error terms must satisfy the conditions $\text{VaR}_u^q(\tau|X_q, X_e) = 0 \quad \text{and} \quad \text{ES}_u^e(\tau|X_q, X_e) = 0$.

Adhering to the methodologies outlined in Lyócsa et al. (2024) and Dimitriadis and Bayer (2019), we apply both M-estimation and Z-estimation to derive the regression parameters and their asymptotic behaviors. Notably, in line with Dimitriadis and Bayer (2019) and Lyócsa et al. (2024) we find the Z-estimator to be unstable, leading to the adoption of the more robust M-estimator. The selected loss function for the estimation process sets $G_1(z) = 0$ and $G_2 = -\log(-z)$ for $z < 0$, as detailed in Dimitriadis and Bayer (2019), facilitating stability and reliability of estimates by ensuring appropriate handling of negative values.

3.2.4. Machine learning

Machine learning algorithms have received considerable attention as alternatives to traditional econometric models for out-of-sample predictions. Athey and Imbens (2019) alludes to their flexible, non-parametric nature. More specifically, their potential ability to handle high-dimensional and correlated predictors with complex non-linear dependencies. A vast number of machine learning algorithms have been proposed. For supervised regressions problems, Athey and Imbens (2019) distinguish between regularized linear regressions (where models are linear in the covariates), regression trees and forests (which are based on partitioning the covariate space), and deep learning (which are constructed by layers of neurons, possibly fully connected through non-linear activation functions). We consider

regularized linear regressions less appropriate for the purpose of this study, as they would require apriori specifications of interactions between IV, RR, and VRP across quantiles. Tree-based methods and neural networks lend themselves to learning these unknown relationships.

CatBoost (Dorogush, Ershov and Gulin, 2018) and XGBoost (Chen and Guestrin, 2016) are gradient boosting algorithms, specifically designed to work without extensive pre-processing and tuning. These models are updated iteratively by adding new trees that aim to reduce the errors of previous iterations:

$$F_t(x) = F_{t-1}(x) + \alpha \cdot h_t(x) \tag{16}$$

$F_t(x)$ represents the model after adding the t-th tree, $F_{t-1}(x)$ is the model up to the $t-1$-th tree, α is the learning rate, and $h_t(x)$ represents the prediction of the t-th tree. Each tree $h_t(x)$ is added to the ensemble to specifically minimize residual errors, enhancing the accuracy of subsequent predictions.

LightGBM (Ke, Meng, Finley, Wang, Chen, Ma, Ye and Liu, 2017) constructs trees in a leaf-wise manner, where the split based on the leaf with the highest potential for reducing loss, unlike the traditional level-wise approach of tree growth used in XGBoost and CatBoost. Mathematically, this process is represented as:

$$\Delta L(\theta) = -\left(\sum_{i \in I} g_i \theta + \frac{1}{2} \sum_{i \in I} h_i \theta^2 \right), \tag{17}$$

where θ is a vector of parameters, $\Delta L(\theta)$ is the change in loss and I denotes the set of data points in a leaf, with g_i and h_i representing the gradients and second-order derivatives of the loss function for each data point i.

LSTM networks (Hochreiter, 1997) are recurrent neural networks designed to process sequential data. The LSTM architecture consists of consecutive layers which collectively enable the network to take the temporal dimension of predictors and target variable into account. Regularization via dropout layers is common to avoid overfitting. Recent financial applications, among many others, include return forecasting (Feng, He and Polson, 2018), asset pricing (Chen, Pelger and Zhu, 2024), option valuation (Pimentel, Risstad, Rogde, Stegavik, Vinje, Westgaard and Wu, 2025; Vinje, Stegavik, Wu, Risstad, Pimentel, Westgaard and Ewald, 2025), risk premia estimation (Rad, Low, Miffre and Faff, 2023), and tail risk assessments (Blom et al., 2023).

~~Section [redacted] contains further details on our specific application of the machine learning models, including final hyperparameters.~~ See Appendix B for further technical details about the machine learning models.

3.3. Forecasting procedure and model tuning

We use a rolling window estimation approach with a window size of 1500 days. For the machine learning models, we use ~~a~~ an 80-20 training-validation split, which leaves 1200 observations for training and 300 days for ~~testing~~

validation. This leads to 1781 estimation windows, spanning from May 5, 2014, to December 28, 2022. At each iteration, we dynamically re-estimate the models and generate one-day-ahead out-of-sample predictions.

To address potential issues of quantile crossing, we implement a correction strategy similar to that proposed by Lyócsa et al. (2024). Specifically, if a predicted quantile value for VaR or ES at a higher level is found to be lower than a corresponding value for a lower quantile, we resolve this inconsistency by adjusting the higher quantile value to be marginally greater than the lower one, with an increment of 0.0001. This adjustment maintains the logical consistency and order of our quantile forecasts for both VaR and ES throughout the forecasting period. We apply this correction across all model specifications.

For machine learning models hyperparameter tuning we use the first estimation window and fix the resulting hyperparameters for the remainder of the analysis. Weights and biases, however, are updated as part of the dynamic model retraining at subsequent rolling windows. Christensen, Siggaard and Veliyev (2023) note that no theoretical consensus has emerged with regard to optimal hyperparameter tuning in the context of financial risk measurement. Notably, Christensen et al. (2023) report that even un-tuned nonlinear machine learning algorithms provide reasonable VaR estimates. Hence, to balance computational cost and precision, we use the same hyperparameter configuration across specifications and quantiles for each of the machine learning models described in Section 3.2.4. [8]

To arrive at these hyperparameter estimates, we conduct an initial grid search for the most parsimonious specification using an integrated method, which systematically optimizes parameters through an unconstrained cross-validated grid search.[9] The hyperparameter spectrum resulting from the initial grid search, reported in Table B.5, serves as a starting point for further refinement. In the second phase, hyperparameters undergo validation through the unconditional coverage test of Kupiec et al. (1995), the conditional coverage test of Christoffersen (1998), and the dynamic quantile test of Engle and Manganelli (2004).[10] This is a manual iterative process, where the objective is to find hyperparameters that provide robust results across specifications and quantiles for a given model.[11]

To prevent the machine learning models from overfitting, we take several measures. First, we employ L2 regularization to impose penalties on large coefficients. Additionally, early stopping refines our training process by stopping the training when the validation metrics stop improving. Specifically for the LSTM model, we incorporate an exponentially decaying learning rate that accelerates learning initially and gradually reduces adjustments as training progresses, aiming to balance rapid convergence with precise model performance. Furthermore, the LSTM model is

[8] Machine learning models generally allow for unique hyperparameters for each quantile. However, given the number of models, specifications, and quantiles involved - in combination with our approach to approximate ES in Equation 6 - this is practically infeasible.

[9] For the ensemble methods, the `GridSearchCV` module from `Scikit-learn` is utilized. Conversely, the LSTM network employs `Keras's GridSearch Tuner`, tailored specifically for neural network configurations.

[10] See Appendix C for definitions.

[11] We employ an alterantive and purely data-driven hyperparameter tuning process in subsection B.6, which confirms the validity of our main results.

Table 3
Finalized hyperparameter architecture for the ensemble methods.

Parameter[a]	CatBoost	LightGBM	XGBoost
Max depth	4	–	4
Number of boosting rounds	400	400	400
Early stopping	50	50	20
Learning rate	0.08	0.008	0.03
Loss function	Quantile	Quantile	Quantile
L2 regularization strength	3	1	1

[a] For all other parameters, default values from the `catboost`, `lightgbm`, and `xgboost` Python libraries are used.

Table 4
Finalized hyperparameter architecture for the LSTM neural network.

Parameter[a]	Value
LSTM architecture	
Layers	3
Units per layer	128, 64, 1
Dropout layers	2
Dropout rate	0.2
Recurrent dropout	0.1
Activation function	ReLU
Training configuration	
Epochs	100
Batch size	16
Early stopping	15
Optimizer	Adam
Initial learning rate	0.01
Learning rate decay	0.96

[a] For all other parameters, default values from the `Keras` Python libraries are used.

subject to dropout. The performance of the LSTM model is continuously evaluated using a loss function that quantifies prediction errors, with weight adjustments made accordingly to optimize training results.

3.4. Expected shortfall forecast evaluation

~~Next, we present the two approaches we use to evaluate the ES predictions. First, we implement the Strict Expected Shortfall Regression (ESR) backtest, as developed by~~ Bayer and Dimitriadis (2022). ~~In addition, we compute a set of loss functions and rank the models using the Model Confidence Set of~~ Hansen, Lunde and Nason (2011).

To evaluate the ES predictions, we compute a set of loss functions and rank the models using the Model Confidence Set of Hansen et al. (2011).[12]

3.4.1. Statistical comparison using loss functions

~~Even though the backtest above might distinguish well-specified from ill-specified models, it does not assess the accuracy of competing models.~~ To obtain a ranking of models, we employ two loss functions specifically designed to jointly assess the accuracy of VaR and ES. The first loss function is the FZ loss function, as specified by Patton, Ziegel

[12] In Appendix A we also employ the Strict Expected Shortfall Regression (ESR) backtest, as developed by Bayer and Dimitriadis (2022), to distinguish well-specified from ill-specified models.

and Chen (2019). The function takes the realized return, VaR and ES as input for each prediction in each model for the six quantiles of interest. The function is defined as:

$$FZ_t^{(0)} = \frac{L_t(R_t - VaR_t)}{\tau ES_t} + \frac{VaR_t}{ES_t} + \log(-ES_t) - 1, \tag{18}$$

where L_t is an indicator function that activates if $R_t < VaR_t$. This formulation allows us to penalize inaccuracies in dual forecasts effectively, evaluating each prediction across multiple quantiles.

Furthermore, we adopt a second loss function, which is a function based on Asymmetric Laplace (AL) density, as introduced by Taylor (2019). The AL loss function is given by:

$$AL_t = -\log\left(\frac{\tau - 1}{ES_{R_t}}\right) - \frac{(R_t - VaR_t)(\tau - L_t)}{\tau ES_{R_t}}, \tag{19}$$

where τ is the probability level for the quantile forecast.

3.4.2. Model Confidence Set

To evaluate the competing models and specifications, we rely on the Model Confidence Set (MCS) of Hansen et al. (2011). The MCS identifies a set of models with equivalent predictive ability that outperform all the other competing models at a given confidence level. The objective of the MCS procedure is to identify the optimal subset of models, M^*, from an initial set of competing models, M^0, at a predefined confidence level. $\hat{M}^* \subset M^0$ will encompass the models $M \in M^0$ that demonstrate the strongest relative forecasting performance according to a specific loss function. This method does not require prespecifying a preferred benchmark model; in fact, it is a statistical test of equivalence with respect to a particular loss function. The trimming is achieved via a sequence of equal predictive ability (EPA) tests. Hence, if the null hypothesis is rejected, the model with the poorest performance is removed from M. This sequential testing procedure continues until the null hypothesis of equal predictive ability is accepted at the given significance level and the Superior Set Models (SSM) $\hat{M}^*$ is obtained.

4. In-sample estimation results

~~We begin by considering full in-sample estimation results from the~~ Dimitriadis and Bayer (2019) model. ~~This enables us to observe the coefficients for both VaR and ES for different quantiles and have an understanding about the importance and directions of the RR and VRP. Furthermore, by including interaction terms, the results also illustrate the interplay between RR and VRP.~~

Table 5
In-sample regression coefficients

Panel A: VaR

	Without Interactions						With Interactions					
	1.0%	**2.5%**	**5.0%**	**95.0%**	**97.5%**	**99.0%**	**1.0%**	**2.5%**	**5.0%**	**95.0%**	**97.5%**	**99.0%**
β_0	-1.567***	-1.211***	-0.922***	0.923***	1.148***	1.469***	-1.523***	-1.178***	-0.934***	0.929***	1.144***	1.469***
β_{IV}	-0.218***	-0.171**	-0.103**	0.050*	0.039	0.114*	-0.166**	-0.153***	-0.108***	0.049*	0.030	-0.075
β_{RR}	0.013	0.072	0.039	-0.011	0.000	-0.017	0.005	0.05	-0.001	-0.007	0.038	0.000
β_{VRP}	-0.201**	-0.169***	-0.103**	0.057'	0.089'	0.164	-0.204**	-0.168**	-0.142**	0.049*	0.103*	0.177'
$\beta_{IV,RR}$							0.167	0.112	0.0688*	-0.027	-0.028	0.044
$\beta_{IV,VRP}$							-0.010	-0.030	-0.017	-0.028	-0.026	-0.050
$\beta_{RR,VRP}$							-0.066	-0.063	-0.052	0.000	0.058	-0.107
Hit-ratio	1.00%	2.52%	4.98%	96.0%	97.18%	98.80%	1.00%	2.55%	5.00%	95.30%	97.15%	98.90%

Panel B: ES

β_0	-1.826***	-1.5353***	-1.289***	1.293***	1.563***	1.982***	-1.824***	-1.517***	-1.281***	1.270***	1.540***	1.967***
β_{IV}	-0.223*	-0.173**	-0.134**	0.050	0.059	0.100	-0.126	-0.139*	-0.125**	0.044	0.046	-0.087
β_{RR}	0.058	0.055	0.043	-0.003	-0.063	-0.159	-0.067	-0.030	-0.022	0.003	0.017	0.161
β_{VRP}	-0.083	-0.132*	-0.139**	0.142'	0.208'	0.34	-0.166	-0.153'	-0.175**	0.152	0.203'	-0.34
$\beta_{IV,RR}$							0.156	0.123	0.102**	-0.053	-0.058	-0.033
$\beta_{IV,VRP}$							0.069	-0.018	-0.002	-0.042	-0.122	0.124
$\beta_{RR,VRP}$							-0.077	-0.056	-0.058	0.056	0.117	-0.191
p-values	0.83	0.99	0.92	0.95	0.99	0.82	0.62	0.97	0.77	0.86	0.92	0.75

Dimitriadis and Bayer (2019) in-sample regression coefficients with and without interactions. Statistical significance codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '' 0.1 ' ' 1. VaR hit-ratio: Proportion of returns below quantile. ES p-values according to McNeil and Frey (2000).

We begin by considering the full in-sample estimation results from the Dimitriadis and Bayer (2019) model. The signs and magnitudes of the estimated regression coefficients provide insights into the model's economic dynamics. Moreover, by including interaction terms, we obtain an illustration of the interplay between independent variables.

The full spectrum of predictors in our empirical analysis spans IV, RR and VRP with daily, weekly, and monthly time to expiry. To facilitate interpretation of the Dimitriadis and Bayer (2019) in-sample regression coefficients, we employ suitable variable transformations. First, we perform a principal component decomposition (PCA) of IV, RR and VRP. This factor representation retains the information content of the original data while reducing the number of explanatory variables. Second, we use the scores of the first principal component of IV, RR, and VRP as explanatory variables. Note that we standardize the variables to enable a meaningful comparison of the relative importance of explanatory variables. Hence, for the purpose of the in-sample analysis, the vector X' in Equation 15 contains standardized values of the first principal components scores for IV, RR, and VRP.[13]

Table 5 reports regression coefficients for VaR (upper panel) and ES (lower panel), without interaction variables in the left panel and including interaction variables in the right panel. As evidenced by the hit-ratio and by the p-value of the McNeil and Frey (2000) ES test, the regression models for both VaR and ES fit the data well in-sample. The coefficient on IV is negative in the left tail and positive in the right tail. This applies both for VaR and ES and across specifications with or without interactions. This is as expected and supports the interpretation of IV as a measure of overall market risk (Gunnarsson et al., 2024).

[13]The PCA decomposition shows that the first principal component explains more than 90% of the cross-sectional variance for both IV, RR, and VRP. The correlation matrix in Table 2, which reports high block-wise correlations between IV, RR, and VRP, supports this approach. Furthermore, it is consistent with a reasonable a priori assumption that daily, weekly, and monthly IVs are most likely to be impacted by common underlying economic drivers. It also aligns well with the general term-structure literature across markets, see for instance Pimentel, Risstad and Westgaard (2022) and references therein.

~~Considering the models without interaction, the sign of the RR and VRP are consistent across tails and risk measures. For RR the coefficients are negative in the left tail and positive in the right tail, whereas for VRP the coefficients are positive in the left tail and negative in the right tail. The magnitude of coefficients is higher or VRP compared to RR, and generally also statistically significant. This suggests that VRP serves as a correction of the well-known bias in IV as a predictor of empirical volatility. We note that although coefficients on the interaction variables generally are different from zero, few are significant. This might be due to a low signal-to-noise ratio in the true data-generating process. Also, specifying the functional form of interactions apriori is generally challenging, in particular in the context of complex extreme events. Arguably, high variance in tail parameters should be expected given the relatively sparse amounts of tail data. Caution should be exercised in general when interpreting regression coefficients and standard errors where predictors are potentially correlated. Still, the in-sample results support that neither IV, RR, nor VRP should be disregarded as relevant predictors for conditional tail risk.~~

Considering the models without interaction, the VRP coefficients are positive in the left tail, negative in the right tail, and generally statistically significant. Comparing the order of magnitude for IV and VRP suggests that VRP serves as a correction for the well-known bias in IV as a predictor of empirical volatility (Slim et al., 2020). For RR the coefficients are negative in the left tail and positive in the right tail. Compared to IV and VRP the magnitude of the coefficients are lower, and not statistically significant. In isolation, this suggests that RR might be less informative for ES predictions within the DB econometric framework.

A general observation is that, apart from the intercept, relatively few coefficients in Table 5 are statistically significant, in particular for specifications including interactions (right panel). A strict econometric interpretation suggests that these variables should not be expected to consistently help predict EURUSD tail risk. A low signal-to-noise ratio in the true data-generating process would render this interpretation correct. On the other hand, the asymptotic assumptions in robust sandwich-type standard errors might be violated in small samples. Arguably, high variance in tail parameters should be expected given the relatively sparse amounts of tail data. Furthermore, inference on unconditional parameter estimates should generally be performed with caution due to structural breaks. As illustrated in Figure 1, our sample covers several global macroeconomic events, which complicates inference of unconditional models. An alternative approach to investigating the relevance of predictors is conditional out-of-sample analysis, which we will turn our attention to in the following sections.

5. Out-of-sample forecasting results

In this section, we report out-of-sample ES forecast evaluation results using the FZ loss function of Bayer and Dimitriadis (2022) and the AL loss function of Taylor (2019), as outlined in Section 3.4.1. We begin by evaluating the competing models and specifications individually in Section 5.1. We then proceed with a model averaging approach in Section 5.2, to ~~reduce~~ alleviate potential effects of idiosyncratic model uncertainty.

5.1. Individual models

To illustrate the dynamics of the models utilized in this study, Figure 6 visualizes the VaR (blue) and ES (purple) predictions for the 5% and 95% quantiles, along with the true returns (black dots), from the CatBoost IV-RR-VRP specification. The predictions align well to major macroeconomic events, as showcased in Figure 1. The conditional predictions increase in magnitude following the global oil price decline i 2015, the Covid19 outbreak in 2020, the Ukrainian war commencing in 2022 as well as the monetary policy actions taken by most central banks following the surge in global inflation.

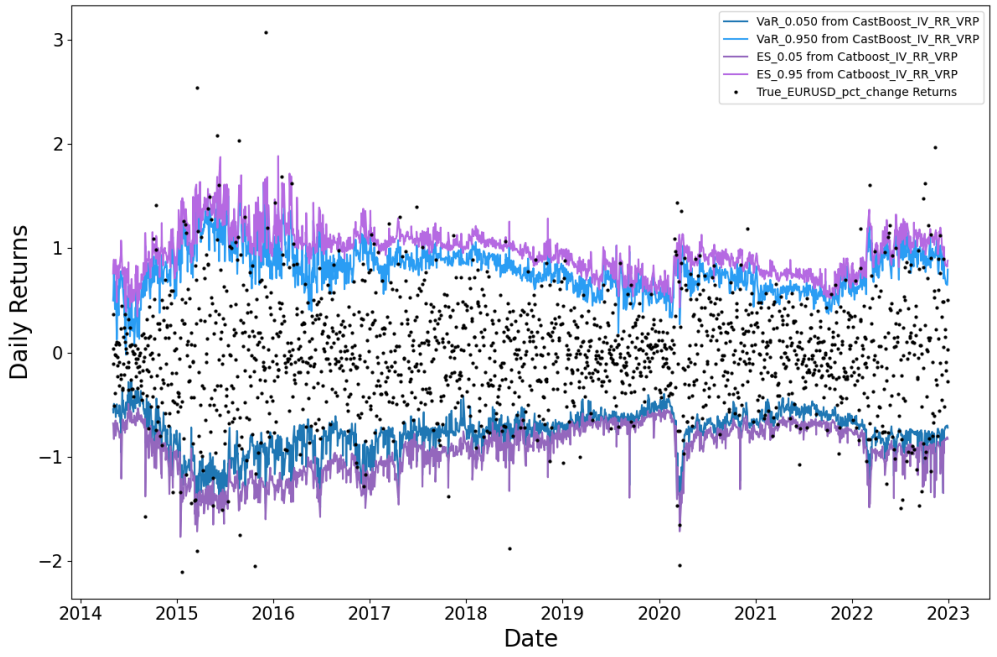

Figure 6: ES and VaR out-of-sample predictions for the CatBoost specification using IV, RR, and VRP as explanatory variables.

Tables 6 and 7 contain results for the FZ and AL loss functions, respectively. In general, results are similar for the two functions across models and quantiles. One notable overall result is that benchmark models are very rarely included in the SSM. The most prominent among the benchmark models appears to be the most parsimonious specification of the linear quantile regression model, QR-IV, which is part of the SSM for some quantiles in both tails. The SSM

is dominated by the tree-based machine learning models and this is particularly pronounced in the left tail. The best-performing model is CatBoost, where all four specifications are included in the SSM for all quantiles. The deep learning LSTM model does generally not perform well, possibly due to misspecification (see specification tests in Appendix A).

Furthermore, an asymmetric pattern emerges: For the machine learning models losses in the left tail are lower than those in the right tail, while the opposite is true for the benchmark models. The number of models retained in the SSM is also generally higher in the left tail. In combination, this indicates the presence of different market dynamics during euro depreciation and appreciation. This again implies that time-varying dynamics might render different models and specifications preferable in different market states. In this respect CatBoost is particularly robust, being part of the SSM for all quantiles, and stands out as the single best-performing model in the right tail.

Even though no clear conclusion can be drawn from the above with respect to the relevance of predictors and related model specifications, some patterns emerge. First, the benchmark models apparently do not benefit from expanding the predictor set. The general trend is that the computed losses increase in both tails as additional predictors beyond IV are introduced. Hence, parsimonious specifications for the benchmark models perform better. For the best-performing machine learning models, the opposite occurs. CatBoost and XGBoost IV-RR-VRP has lower losses in the left tail compared to the corresponding IV specifications. The LSTM model displays similar properties in the right tail, where the IV-RR-VRP specification is among the superior models as evaluated by the AL function.

Table 6
FZ loss, MCS, and average ranks

Model	0.01 ES			0.025 ES			0.05 ES			0.95 ES			0.975 ES			0.99 ES		
	FZ	M*	R	FZ	M*	R	FZ	M*	R	FZ	M*	R	FZ	M*	R	FZ	M*	R
Panel A: Benchmark models																		
EGARCH IV	**0.520**		23	**0.277**		25	**0.078**		24	**0.033**		18	**0.209**		22	**0.455**		24
EGARCH IV-RR	**0.579**		26	0.309		26	**0.100**		25	0.043		24	**0.216**		23	**0.438**		23
EGARCH IV-VRP	**0.508**		22	**0.271**		23	**0.074**		23	**0.031**		16	**0.206**		19	**0.427**		21
EGARCH IV-RR-VRP	**0.583**		27	0.313		27	**0.104**		26	0.040		21	**0.207**		20	**0.420**		19
DB IV	**0.453**		18	**0.185**		16	**0.012**		15	**0.054**		25	**0.235**		25	**0.392**		13
DB IV-RR	**0.533**		24	**0.149**	*	13	0.031		20	**0.080**		27	**0.305**		27	**0.405**		15
DB IV-VRP	0.473		19	**0.190**		17	**0.021**		19	**0.068**		26	**0.292**		26	**0.521**		26
DB IV-RR-VRP	**0.431**	*	17	**0.249**		22	**0.017**		18	**0.042**		23	**0.229**		24	**0.852**		28
QR IV	**0.411**	*	15	0.182	*	15	0.000		13	0.008	*	12	**0.169**	*	5	**0.316**	*	1
QR IV-RR	0.508		21	0.197	*	19	0.011		14	0.039		19	**0.192**		13	**0.337**	*	3
QR IV-VRP	**0.540**		25	0.216		20	0.014		17	0.024		14	**0.207**		21	**0.419**		18
QR IV-RR-VRP	**0.833**		28	0.335		28	0.063		22	0.040		22	**0.194**		14	**0.410**		16
Panel B: Machine learning models																		
CatBoost IV	**0.293**	*	8	**0.102**	*	4	**-0.055**	*	4	-0.047	*	1	**0.120**	*	1	**0.344**	*	6
CatBoost IV-RR	**0.284**	*	7	**0.100**	*	3	-0.060	*	2	**-0.042**	*	2	**0.129**	*	2	**0.326**	*	2
CatBoost IV-VRP	**0.322**	*	11	**0.095**	*	2	**-0.054**	*	5	-0.022	*	4	0.137	*	4	**0.352**	*	4
CatBoost IV-RR-VRP	**0.260**	*	4	**0.072**	*	1	-0.084	*	1	**-0.030**	*	3	**0.133**	*	3	**0.382**	*	10
XGBoost IV	**0.331**	*	12	**0.143**	*	12	**-0.024**		12	0.000		8	0.180		11	**0.375**		9
XGBoost IV-RR	**0.259**	*	3	**0.104**	*	5	**-0.040**		10	-0.003	*	7	**0.177**		9	**0.369**	*	7
XGBoost IV-VRP	**0.305**	*	9	**0.114**	*	8	**-0.041**		9	-0.013	*	5	**0.170**		7	**0.389**		11
XGBoost IV-RR-VRP	**0.309**	*	10	**0.105**	*	6	**-0.053**		6	-0.004	*	6	**0.177**		8	**0.394**		14
LightGBM IV	**0.266**	*	5	**0.122**	*	10	**-0.046**		7	0.006		11	**0.185**		12	**0.375**		8
LightGBM IV-RR	**0.256**	*	2	**0.122**	*	11	-0.038		11	0.006		10	**0.195**		16	**0.362**	*	5
LightGBM IV-VRP	0.255	*	1	0.106	*	7	**-0.056**	*	3	**0.000**	*	9	0.198		16	0.421		20
LightGBM IV-RR-VRP	**0.273**	*	6	**0.122**	*	9	**-0.045**		8	0.009		13	0.199		17	**0.414**		17
LSTM IV	0.347	*	13	0.171		14	0.012		16	0.033		17	0.178		10	0.427		22
LSTM IV-RR	0.420		16	0.252		22	0.139		28	0.027		15	0.205		18	0.492		25
LSTM IV-VRP	0.486		20	0.276		24	**0.110**		27	**0.093**		28	0.343		28	0.797		27
LSTM IV-RR-VRP	0.367		14	0.193		18	0.057		22	0.039		21	0.169		6	0.392		12

The values in the table are the average FZ^0 loss of the ES model for each quantile. Models marked with * belong to the superior set of models, M*, as determined by the MCS procedure at 80% confidence level. We obtain the quantiles of the asymptotic distribution of the test statistic (maximum difference between model performances) by block bootstrap with 5,000 replications and a block length of $\sqrt{T}$. The "R" column indicates the rank of the mean of each model, with 1 being the best average loss. Higher numbers correspond to worse performance. Loss values for models that pass the specification test outlined in Section A at the 5% significance level are reported in bold.

Table 7
AL loss, MCS, and average ranks

Model	0.01 ES			0.025 ES			0.05 ES			0.95 ES			0.975 ES			0.99 ES		
	AL	M*	R	AL	M*	R	AL	M*	R	AL	M*	R	AL	M*	R	AL	M*	R
Panel A: Benchmark models																		
EGARCH IV	**1.530**		23	**1.302**		25	**1.130**		24	**1.126**		19	**1.270**		22	**1.495**		24
EGARCH IV-RR	**1.589**		26	1.334		26	**1.151**		25	1.137		24	**1.278**		23	**1.478**		23
EGARCH IV-VRP	**1.519**		22	**1.296**		23	**1.125**		23	**1.124**		18	**1.267**		20	**1.467**		22
EGARCH IV-RR-VRP	**1.593**		27	1.339		27	**1.156**		26	**1.134**		22	**1.268**		21	**1.461**		21
DB IV	**1.460**		18	**1.210**		16	**1.063**		16	**1.142**		25	**1.291**		25	**1.430**		14
DB IV-RR	**1.543**		24	**1.175**	*	13	1.082		20	**1.173**		28	**1.363**		27	**1.444**		18
DB IV-VRP	1.483		19	**1.215**		17	**1.070**		19	**1.157**		27	**1.346**		26	**1.553**		26
DB IV-RR-VRP	**1.441**	*	17	**1.274**		21	**1.067**		18	**1.136**		23	**1.288**		24	**1.886**		28
QR IV	**1.422**	*	15	1.207	*	15	1.052		13	1.097		15	**1.226**		12	**1.351**	*	2
QR IV-RR	1.518		21	1.223	*	19	1.062		15	1.131		20	**1.251**		17	**1.375**	*	4
QR IV-VRP	**1.547**		25	1.240		20	1.062		14	1.113		17	**1.264**		19	**1.454**		20
QR IV-RR-VRP	**1.844**		28	1.360		28	1.114		22	1.134		21	**1.254**		18	**1.449**		19
Panel B: Machine learning models																		
CatBoost IV	**1.303**	*	8	**1.127**	*	4	**0.996**	*	4	1.022	*	1	**1.158**	*	1	**1.384**	*	6
CatBoost IV-RR	**1.294**	*	7	**1.126**	*	3	0.991	*	2	**1.030**	*	2	**1.167**	*	2	**1.348**	*	1
CatBoost IV-VRP	**1.332**	*	11	**1.121**	*	2	**0.998**	*	5	1.042	*	4	1.172	*	3	**1.372**	*	3
CatBoost IV-RR-VRP	**1.270**	*	4	**1.097**	*	1	0.968	*	1	**1.041**	*	3	**1.172**	*	4	**1.403**	*	10
XGBoost IV	**1.342**	*	12	**1.1690**	*	12	**1.028**		12	1.068		10	1.214		10	**1.392**	*	8
XGBoost IV-RR	**1.269**	*	3	**1.129**	*	5	**1.012**		10	1.072		11	**1.213**		9	**1.387**	*	7
XGBoost IV-VRP	**1.315**	*	9	**1.140**	*	8	**1.010**		9	1.060	*	7	**1.202**		6	**1.405**	*	11
XGBoost IV-RR-VRP	**1.319**	*	10	**1.130**	*	6	**0.999**		6	1.073	*	12	**1.210**		7	**1.410**		13
LightGBM IV	**1.276**	*	5	**1.148**	*	10	**1.006**		7	1.063		10	**1.219**		11	**1.393**	*	9
LightGBM IV-RR	**1.266**	*	2	**1.148**	*	11	1.013		11	1.065		11	**1.232**		14	**1.381**	*	5
LightGBM IV-VRP	1.265	*	1	1.132	*	7	**0.995**	*	3	**1.055**	*	7	1.230		13	1.437		16
LightGBM IV-RR-VRP	**1.283**	*	6	**1.147**	*	9	**1.007**		8	1.067		12	1.235		15	**1.431**		15
LSTM IV	1.357	*	13	1.196		14	1.063		17	1.093		13	1.211		8	1.442		17
LSTM IV-RR	1.430		16	1.277		22	1.191		28	1.093		14	1.242		16	1.511		25
LSTM IV-VRP	1.496		20	1.301		24	**1.162**		27	**1.148**		26	1.370		28	1.806		27
LSTM IV-RR-VRP	1.357		14	1.219		18	1.108		21	1.100		16	1.202	*	5	1.408	*	12

The values in the table are the average AL loss of the ES model for each quantile. Models marked with * belong to the superior set of models, M*, as determined by the MCS procedure at 80% confidence level. We obtain the quantiles of the asymptotic distribution of the test statistic (maximum difference between model performances) by block bootstrap with 5,000 replications and a block length of $\sqrt{T}$. The "R" column indicates the rank of the mean of each model, with 1 being the best average loss. Higher numbers correspond to worse performance. Loss values for models that pass the specification test outlined in Section A at the 5% significance level are reported in bold.

Table 8
Equally weighted average losses across specifications, MCS, and ranks

Model	0.01 ES			0.025 ES			0.05 ES			0.95 ES			0.975 ES			0.99 ES		
	FZ	M*	R	FZ	M*	R	FZ	M*	R	FZ	M*	R	FZ	M*	R	FZ	M*	R
Panel A: FZ loss																		
Benchmark models	0.531		2	0.239		2	0.044		2	0.042		2	0.222	*	2	0.450	*	2
ML models	0.315	*	1	0.138	*	1	-0.017	*	1	0.003	*	1	0.181	*	1	0.414	*	1
Panel B: AL loss																		
Benchmark models	1.541		2	1.265		2	1.095		2	1.134		2	1.218		2	1.487	*	2
ML models	1.325	*	1	1.163	*	1	1.034	*	1	1.067	*	1	1.212	*	1	1.432	*	1

The values in the table are the average FZ and AL loss of the ES model combinations for each quantile. Models marked with * belong to the superior set of models, M*, as determined by the MCS procedure at 80% confidence level. We obtain the quantiles of the asymptotic distribution of the test statistic (maximum difference between model performances) by block bootstrap with 5,000 replications and a block length of $\sqrt{T}$. The "R" column indicates the rank of the mean of each model, with 1 being the best average loss. Higher numbers correspond to worse performance. Loss values for models that pass the specification test outlined in Section A at the 5% significance level are reported in bold.

Table 9
Equally weighted average losses pr specification, MCS, and average ranks

Model	0.01 ES			0.025 ES			0.05 ES			0.95 ES			0.975 ES			0.99 ES		
	FZ	M*	R	FZ	M*	R	FZ	M*	R	FZ	M*	R	FZ	M*	R	FZ	M*	R
Panel A: FZ loss																		
Benchmark models IV	0.462		5	0.214		5	0.030		5	0.032	*	5	0.204	*	4	0.388	*	3
Benchmark models IV-RR	0.540		7	0.218		6	0.048		7	0.054	*	8	0.238		8	0.394	*	4
Benchmark models IV-VRP	0.507		6	0.226		7	0.036		6	0.041	*	7	0.235		7	0.456	*	6
Benchmark models IV-RR-VRP	0.616		8	0.299		8	0.062		8	0.041	*	6	0.210	*	5	0.561	*	8
ML models IV	0.310	*	3	0.135	*	2	-0.028	*	2	-0.002	*	2	0.166	*	1	0.385	*	1
ML models IV-RR	0.305	*	2	0.145	*	3	0.000		4	-0.003	*	1	0.177	*	3	0.387	*	2
ML models IV-VRP	0.342		4	0.148		4	-0.010		3	0.015	*	4	0.212		6	0.499		7
ML models IV-RR-VRP	0.302	*	1	0.123	*	1	-0.031	*	1	0.004	*	3	0.170	*	2	0.396	*	5
Panel B: AL loss																		
Benchmark models IV	1.472		5	1.240		5	1.081		5	1.128		5	1.262	*	4	1.425	*	4
Benchmark models IV-RR	1.550		7	1.244		6	1.099		7	1.147		8	1.297		8	1.436	*	5
Benchmark models IV-VRP	1.516		6	1.250		7	1.086		6	1.132		6	1.292		7	1.491	*	6
Benchmark models IV-RR-VRP	1.626		8	1.324		8	1.113		8	1.135		7	1.270		5	1.599		8
ML models IV	1.320	*	3	1.160	*	2	1.023	*	2	1.061	*	1	1.200	*	1	1.402	*	1
ML models IV-RR	1.315	*	2	1.170	*	3	1.052		4	1.065	*	2	1.214	*	3	1.407	*	2
ML models IV-VRP	1.352		4	1.173		4	1.041		3	1.075	*	4	1.248		6	1.505		7
ML models IV-RR-VRP	1.312	*	1	1.148	*	1	1.020	*	1	1.068	*	3	1.205	*	2	1.413	*	3

The values in the table are the average FZ and AL loss of the ES model combinations for each quantile. Models marked with * belong to the superior set of models, M*, as determined by the MCS procedure at 80% confidence level. We obtain the quantiles of the asymptotic distribution of the test statistic (maximum difference between model performances) by block bootstrap with 5,000 replications and a block length of $\sqrt{T}$. The "R" column indicates the rank of the mean of each model, with 1 being the best average loss. Higher numbers correspond to worse performance. Loss values for models that pass the specification test outlined in Section A at the 5% significance level are reported in bold.

5.2. Model averages

Comparisons across a broad set of models and specifications, as reported in Section 5.1, rarely identify one single superior competitor. Hence, to diversify idiosyncratic model risk and reach more general conclusions, we compute model averages.

Table 8 reports results for equally weighted averages across all four specifications, for benchmark models and machine learning models respectively. This allows us to evaluate whether machine learning models in general are more efficient in exploiting the combined predictive information of IV, RR, and VRP. The average losses are lower for the machine learning models across all quantiles, both for the AL and FZ functions. The MCS procedure generally renders these differences as statistically significant, except for the far right tail. From this we conclude that the machine learning models are preferred over the benchmark models, when evaluated as consolidated model classes.

To increase granularity and shed further light on the relevance of different predictors, Table 9 reports equally weighted averages for each of the four specifications. The results reinforce that machine learning models are generally preferred over benchmark models, also when alternative specifications are controlled for. In the left tail, we note

that the ML IV-RR-VRP specification delivers the lowest average losses. In the right tail the more parsimonious ML IV specification performs best on average, while IV-RR-VRP is still ranked among the better specifications. We note, however, that these differences are statistically insignificant in the MCS procedure. Furthermore, the IV-VRP specification is generally not included in the MCS. In combination, this suggests that although VRP appears superfluous in isolation, it is arguably still relevant as a predictor when combined with IV and RR. In the right tail, the MCS procedure is less able to distinguish between the competing specifications. The average difference in performance between machine learning models and benchmark models is also less pronounced.

6. Discussion

6.1. Econometric models vs. ML models and the relevance of predictors

A robust conclusion is that the ML models generally outperform the econometric benchmark models. This is evident on an aggregate level and when comparing individual models and specifications. Table 8, which contains equally weighted results across all specifications, shows that the average loss is lower for ML models in all quantiles. These differences are generally statistically significant, except for the extreme right tail of the distribution. Hence, the flexible and non-parametric nature of ML models leads to more accurate tail risk estimates.

To determine whether more complex specifications outperform parsimonious specifications, Table 9 reveals different dynamics for econometric and ML models. For the benchmark models, the IV specification has the lowest average loss for all quantiles and both the AL and FZ loss functions. In most cases, IV-RR-VRP has the highest average loss among the four specifications. Table 6 and Table 7 confirm that this general trend applies also at the individual model level. In total, this suggests that the econometric models are unable to benefit from expanding the set of predictors beyond IV. The ML models display a different pattern. In the left tail, IV-RR-VRP has the lowest average loss, whereas IV has the lowest average loss in the right tail. This asymmetry indicates that the appropriate specification is conditional on the prevailing market regime. Furthermore, we observe that the IV-VRP specification is never included in the SSM, while IV-RR is included for both tails. This is somewhat contrary to the DB in-sample estimates in Section 4. Still, average out-of-sample losses in Section 5 indicate that combining RR and VRP in the IV-RR-VRP specification is optimal. However, both IV-RR-VRP and IV are included in the SSM when averaged over the full out-of-sample period.

6.2. Loss differentials over time

Figure 7 displays the cumulative losses from the simplest and the most complex variants of CatBoost and EGARCH for the 0.01 quantile. As observed, the CatBoost model consistently exhibits higher losses for its simplest variant over an extended period, as indicated by the dark blue line surpassing the lighter one between 2015-2018. However, this pattern is disrupted in 2018 with a increase in losses where the complex variant surpasses the simpler one. Notably, a sudden increase is also evident in 2020, coinciding with the onset of the COVID-19 pandemic, slightly more evident in the IV variant. A similar increase in 2020 is observed for the EGARCH model, but with a more pronounced magnitude. The visible difference between the cumulative loss of the two model variants of the EGARCH model suggests higher impact on the most complex variant. The losses for the EGARCH variants displays the same dynamics with similar values until this significant increase in 2020. This suggest that the EGARCH model have not adjusted its prediction behavior when utilizing the additional variables, RR and VRP, in this period. The y-axis displays the difference in absolute cumulative loss between the CatBoost and EGARCH models, which is significant in terms of both its incremental

growth and sudden increases. This significant divergence aligns with the high relative performance of CatBoost in the MCS procedure and its favorable FZ-loss ranking, as observed in Table 6.

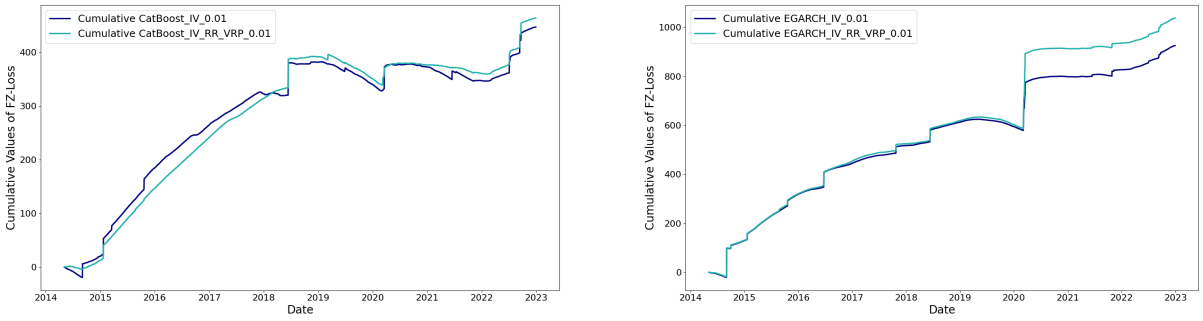

Figure 7: Comparison of CatBoost and EGARCH cumulative FZ losses for the 0.01 ES levels utilizing IV, and IV-RR-VRP.

We observe some discrepancies between the results of the loss function-based evaluations presented in Section 5 and the strict ESR backtest presented in Appendix A. Specifically, the strict ESR backtest highlights potential issues for several machine learning models to produce well-specified ES forecasts at higher quantiles. This suggests difficulties in accurately predicting extreme scenarios where the euro becomes stronger relative to the US dollar. Conversely, the loss functions and the MCS procedure suggests high degree of accuracy in the predictions for the machine learning models compared to the GARCH-class models in the upper tail.

At the 0.975 quantile, we observe rejection in the strict ESR backtest for CatBoost-IV and CatBoost-IV-VRP, but never for any EGARCH variants. By looking at the cumulative losses for this quantile in Figure 8, we observe some discrepancies that might also underscore the results from the strict ESR backtest. Specifically, the cumulative losses of CatBoost displays a decrease between 2019-2022. This is likely a consequence of predictions not achieving the appropriate amount of breaches, thus resulting in negative loss. The EGARCH models also exhibit periods of negative losses after 2019, although less pronounced. In other words, this indicates that CatBoost models are not assessing the risk properly, which can explain the evidence of model misspecification in Table A1. Furthermore, CatBoost-IV consistently achieves lower losses throughout the entire period compared to CatBoost-IV-VRP, justifying its ranking as the top-performing model. Since negative losses are not treated differently in this evaluation, they will offset positive losses ~~and thus reduce~~, thereby reducing the overall reported loss for a model.

In order to assess the generality of our assessments from Figure 7 and Figure 8, it is beneficial to see if similar patterns emerge for different quantiles and loss functions. Therefore, Figure 9 plots the cumulative loss for the AL function for the 0.025 quantile, again using the simplest as well as the most complex variant of CatBoost and EGARCH. The cumulative loss from this loss function also exhibits periods of abrupt increases, akin to those observed when employing the FZ-loss function. This is more evident for EGARCH compared to CatBoost. Additionally, the difference in cumulative loss between model variants for CatBoost over time suggests that the IV-RR-VRP variant consistently

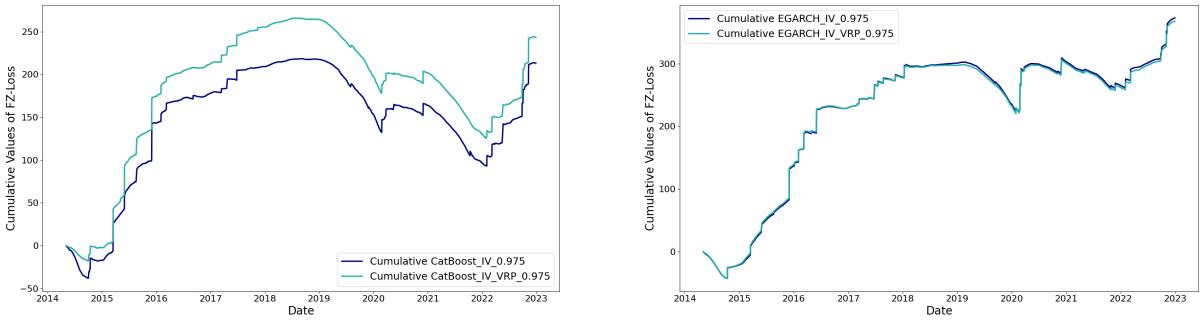

Figure 8: Comparison of CatBoost and EGARCH cumulative FZ losses for the 0.975 ES levels utilizing IV, and IV-VRP.

performs better in terms of accuracy. This substantiates the improvement of forecasting accuracy by incorporating additional variables. For EGARCH, we observe approximately indistinguishable loss between the model variants until the sudden increase in 2020. After this, the difference in cumulative loss is constant, and models achieve similar daily losses again. The observed differences in cumulative loss between the EGARCH-IV and EGARCH-IV-RR-VRP suggest that the rejection in the strict ESR backtest for the most complex variant might be attributed to its greater sensitivity to this unforeseen event.

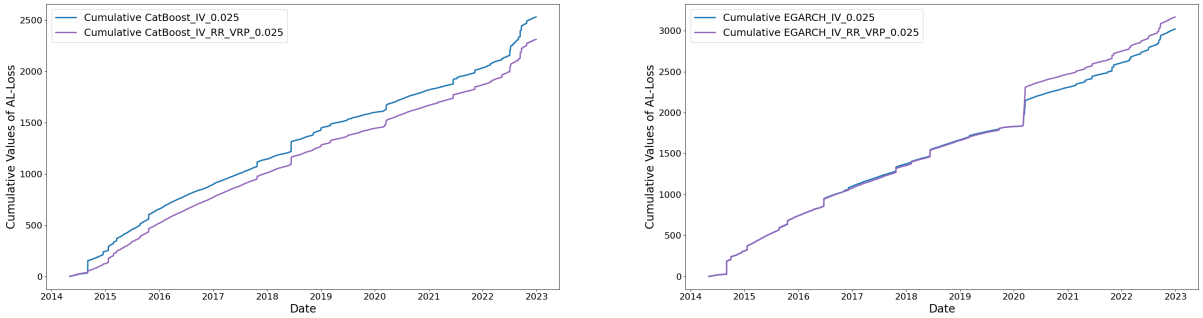

Figure 9: Comparison of CatBoost and EGARCH cumulative AL-losses for the 0.025 ES levels and model variants utilizing IV, and IV-RR-VRP.

6.3. Asymmetric tail risk dynamics

The out-of-sample results in Section 5.1 imply that the benchmark models are unable to benefit from expanding the set of predictors beyond IV. Furthermore, the relevance of RR and VRP as predictors of tail risk in machine learning models apparently depend on whether the euro depreciates or appreciates. To investigate this further, we compute accumulated losses for equally weighted estimates from the machine learning models over the out-of-sample period and analyze the difference between the IV and IV-RR-VRP specifications.

Figure 10 displays the behavior of loss differentials at comparable quantiles over time. Loss differentials above zero indicate that IV-RR-VRP outperforms the more parsimonious IV specification. We note that IV-RR-VRP consistently performs better than IV for all three quantiles in the left tail. Results for the right tail are more mixed. For the 99%

percentile IV is preferred over IR-RR-VRP. For the 97.5% percentile, however, the IV-RR-VRP performs better in the first part of the out-of-sample period. Similarly, for the 95% percentile, the two specifications yield comparable results in the first portion of the sample. For both the 95% percentile and 97.5% percentile, the accumulated loss differentials shift sign approximately at the beginning of 2020, whereby IV outperforms IV-RR-VRP. While refraining from strict causal interpretations, we note that this coincides with the Covid19-outbreak, and it is conceivable that this event constituted a structural EURUSD break, given related implications for international trade and macroeconomics.

The EURUSD spotrate dynamics as displayed in Figure 1 form a natural extension to this discussion. During the first part of our out-of-sample period, the EURUSD was relatively stable, trading in a range around 1.1. The last part of the out-of-sample period, on the other hand, is mainly characterized by a sharp depreciation of the euro, largely driven by major global events and changes in the monetary policies of the Federal Reserve and the European Central Bank.[14] Judging from Figure 10, the IV-RR-VRP specification consistently outperforms IV for conditional left tail ES estimates. In the right tail, however, it is not straightforward to distinguish between the two. Notably, in the right tail, the more parsimonious IV specification appears to be more favorable when the euro is depreciating.

Lastly, we note that these interpretations are robust, in that AL and FZ losses are highly correlated.

[14] Between 2020 and 2023, the EURUSD exchange rate experienced notable fluctuations driven by macroeconomic and geopolitical factors. The onset of the COVID-19 pandemic in 2020 initially strengthened the U.S. dollar due to safe-haven demand, followed by a depreciation driven by expansive Federal Reserve monetary policy. In 2022, the exchange rate declined sharply, reaching parity for the first time in two decades, as the Federal Reserve implemented aggressive rate hikes in response to surging inflation, while the euro was pressured by the economic fallout from the Russia-Ukraine conflict. In 2023, the euro regained strength as the European Central Bank adopted a more hawkish stance and market expectations shifted toward a potential deceleration in U.S. monetary tightening. Overall, the EURUSD exchange rate dynamics during this period were predominantly shaped by interest rate differentials, inflation trends, and geopolitical instability.

Figure 10: Cumulative loss differentials. IV-RR-VRP vs IV for equally weighted machine learning model results. AL loss (left panel) and FZ loss (right panel). Right (left) tail quantiles in blue (red). Positive numbers imply that IV-RR-VRP outperforms IV.

6.4. Feature importance

The linear quantile regression model has proven to be accurate for capturing tail risk across asset classes, also for EURUSD (Haugom et al., 2016; de Lange et al., 2022; Lyócsa et al., 2024). In this study, however, we find that machine learning models are superior. In an attempt to infer why this is the case, we analyse feature importance for CatBoost and the linear quantile regression model, utilizing both IV, RR, and VRP. We focus on the period from 2020 to 2023, as this contains significant tail events.

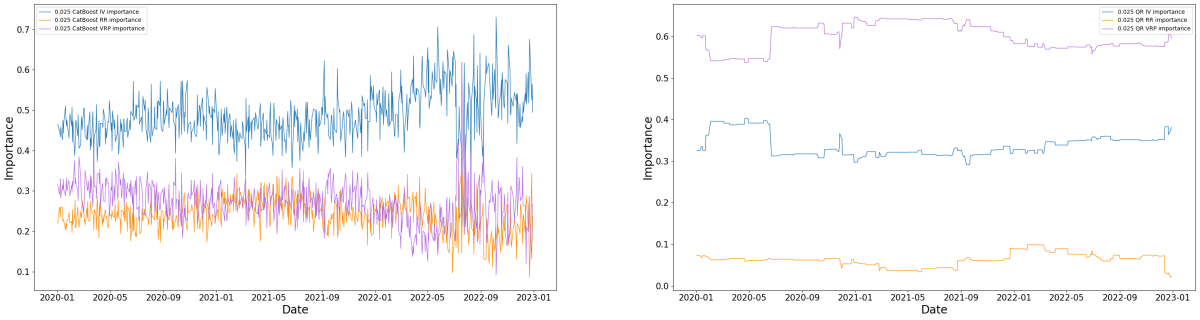

Figure 11: Variable importance of CatBoost and QR for the 0.025 VaR for the period 2020-2023.

In Figure 11, we observe that the CatBoost model displays a high degree of short-term variation in feature importance, even though the relative weighting appears relatively stable over time, indicating structural stability. Another notable observation is the relative ranking of the predictors. For CatBoost, IV is the most important class of predictors. This is reasonable, since IV is often interpreted as a measure of overall market risk. VRP, serving as a correction for bias in IV, and RR, capturing the probability of tail events, are both less important. We interpret this as an indication of CatBoost being able to learn the complex, non-linear, and time-varying interactions between the set of correlated predictors. The quantile regression model, on the other hand, exhibits very limited variability in relative feature importance. Given the rolling window estimation approach, which is designed to enhance adaptability to changing market conditions, and the significant market events contained in this subsample, a reasonable a priori assumption would be that there is some variance in feature importance. The quantile regression model is likely to suffer from its linear formulation in combination with multicollinearity.

~~In~~ Figure 11, ~~we observe that the quantile regression model exhibits very limited variability in relative feature importance. Given the rolling window estimation approach, which is designed to enhance adaptability to changing market conditions, and the significant market events contained in this subsample, a reasonable apriori assumption would be some variance in feature importance. Surprisingly, VRP appears to be most important in the quantile regression model, while RR is close to negligible. In total, we interpret this as indications of CatBoost being able to learn the complex, non-linear and time-varying interactions between the set of correlated predictors. The quantile regression model, on the other hand, most likely suffers from its linear formulation in combination with multicollinearity.~~

6.5. Comparison of tree-based methods

Figure 12 displays the difference in cumulative FZ losses for CatBoost IV-RR-VRP compared to XGBoost (left) and LightGBM (right). The results are similar in that CatBoost consistently performs better in the right tail over the out-of-sample period. Although less pronounced, CatBoost appears to be more accurate also in the left tail. A likely explanation is that CatBoost is less sensitive to multicollinearity compared to other boosting methods. CatBoost uses oblivious trees. This involves growing symmetric trees, with the same condition on both left/right splits at each level. This reduces the complexity of the tree and leads to more general patterns being learned by decreasing the likelihood of overfitting by repeatedly choosing correlated splits. Furthermore, CatBoost employs regularization via target encoding with noise and prior smoothing, which helps to stabilize the contribution of variables. This is of particular relevance in this analysis, where blocks of independent variables are highly correlated, as discussed in Section 2.

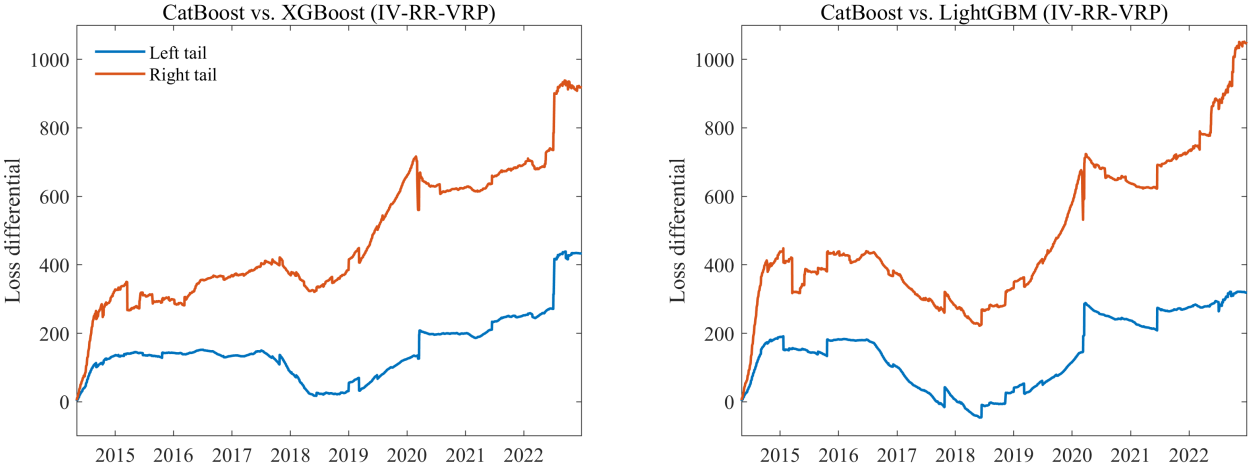

Figure 12: Cumulative FZ loss differentials, IV-RR-VRP specification. Left tail: 1%, 2.5%, and 5% quantile losses. Right tail: 95%, 97.5%, and 99% quantile losses. CatBoost vs. XGBoost (left) and CatBoost vs. LightGBM (right). Positive numbers imply CatBoost outperformance.

7. Conclusion

In this paper, we explore a variety of econometric and machine learning models to estimate expected shortfall for daily EURUSD returns. We build on the heterogeneous market participants hypothesis and use implied volatilities from options with daily, weekly, and monthly time-to-expiry. Extensive out-of-sample analysis highlights several robust insights.

First, tree-based ensemble models consistently outperform econometric benchmarks. Their ability to flexibly accommodate high-dimensional, collinear predictors and nonlinear dynamics allows them to adjust more effectively to regime shifts, especially during turbulent periods such as the COVID-19 crisis and the onset of the war in Ukraine. In

contrast, econometric models perform best in parsimonious specifications, underscoring the limited benefit of adding complex predictors to traditional frameworks. We find that LSTM is less effective.

Second, model performance is asymmetric in the two tails of the conditional return distribution. The ML models, on average, yield lower losses in than the econometric models in the right tail, albeit at comparable performance levels. In the left tail however, during euro depreciation, the difference in performance is much more pronounced. The left tail dynamics are harder to capture for most models and specifications, yet this is precisely where machine learning methods demonstrate the most significant relative improvements.

Third, the evidence regarding the predictor set is less conclusive. Variance risk premia emerge as the most informative addition beyond implied volatility, effectively correcting for the well-documented bias in implied volatilities as predictors of realized variance. Risk reversals provide additional, albeit weaker, predictive content, particularly when embedded in nonlinear models. As evaluated by FZ and AL average losses, CatBoost-IV-RR-VRP emerges as the top performer in the left tail, whereas the more parsimonious CatBoost-IV is preferred in the right tail. These differences are however, not statistically significant when evaluated in the MCS framework. A comparison of cumulative losses for the tree-based models over the out-of-sample period, however, indicates that the IV-RR-VRP specification exhibits preferable dynamics over the parsimonious IV specification, especially in the left tail. Our overall findings suggest that simpler models may be preferred in stable market environments, whereas complex models with high-dimensional predictors may be better suited to capture rapidly changing market conditions.

To this end, we suggest exploring more advanced model averaging techniques, such as the dynamic model averaging method introduced by Koop and Korobilis (2012), or the recently proposed score-driven approach by Fuentes, Herrera and Clements (2025). Similarly, further exploration of feature engineering from the implied volatility surface might prove beneficial. Last but not least, extensions to other currency pairs and longer horizons would provide valuable information.

References

Acerbi, C., Tasche, D., 2002. Expected shortfall: A natural coherent alternative to value at risk. Economic Notes 31. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/1468-0300.00091, doi:https://doi.org/10.1111/1468-0300.00091.

Andersen, T.G., Fusari, N., Todorov, V., 2020. The pricing of tail risk and the equity premium: Evidence from international option markets. Journal of Business & Economic Statistics 38, 662–678.

Athey, S., Imbens, G.W., 2019. Machine learning methods that economists should know about. Annual Review of Economics 11, 685–725.

Barndorff-Nielsen, O., Hansen, P., Lunde, A., Shephard, N., 2008. Realized kernels in practice: Trades and quotes. Econometrics Journal 12. doi:10.1111/j.1368-423X.2008.00275.x.

Basel Committee on Banking Supervision, 1996. Supervisory framework for the use of "backtesting" in conjunction with the internal models approach to market risk capital requirements .

Basel Committee on Banking Supervision, 2016. Minimum capital requirements for market risk. Consultative Document .

Bayer, S., Dimitriadis, T., 2022. Regression-based expected shortfall backtesting. Journal of Financial Econometrics 20, 437–471. URL: https://academic.oup.com/jfec/article/20/3/437/5912157, doi:10.1093/jjfinec/nbaa013.

Bekaert, G., Hoerova, M., 2014. The VIX, the variance premium and stock market volatility. Journal of Econometrics 183, 181–192.

Blom, H.M., de Lange, P.E., Risstad, M., 2023. Estimating value-at-risk in the eurusd currency cross from implied volatilities using machine learning methods and quantile regression. Journal of Risk and Financial Management 16, 312.

Bollerslev, T., 1987. A conditionally heteroskedastic time series model for speculative prices and rates of return. The Review of Economics and Statistics , 542–547.

Bollerslev, T., Tauchen, G., Zhou, H., 2009. Expected stock returns and variance risk premia. The Review of Financial Studies 22, 4463–4492.

Brownlee, J., 2017. XGBoost with Python - Gradient Boosted Trees with XGBoost and scikit-learn. URL: https://books.google.no/books?hl=no&lr=&id=HgmqDwAAQBAJ&oi=fnd&pg=PP1&dq=jason+brownlee+xgboost&ots=nNeHj8LbIG&sig=Z8dsDCG4LROPFbMlkYGR-dgCF9Y&redir_esc=y#v=onepage&q=jason%20brownlee%20xgboost&f=false.

Busch, T., Christensen, B.J., Nielsen, M.Ø., 2011. The role of implied volatility in forecasting future realized volatility and jumps in foreign exchange, stock, and bond markets. Journal of Econometrics 160, 48–57.

Chen, L., Pelger, M., Zhu, J., 2024. Deep learning in asset pricing. Management Science 70, 714–750.

Chen, T., Guestrin, C., 2016. XGboost: A scalable tree boosting system, in: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pp. 785–794.

Christensen, K., Siggaard, M., Veliyev, B., 2023. A machine learning approach to volatility forecasting. Journal of Financial Econometrics 21, 1680–1727.

Christoffersen, P., Mazzotta, S., 2005. The accuracy of density forecasts from foreign exchange options. Journal of Financial Econometrics 3, 578–605.

Christoffersen, P.F., 1998. Evaluating interval forecasts. International Economic Review 39, 841–862. URL: https://doi.org/10.18637/jss.v027.i03, doi:https://doi.org/10.2307/2527341.

Chronopoulos, I., Raftapostolos, A., Kapetanios, G., 2024. Forecasting value-at-risk using deep neural network quantile regression. Journal of Financial Econometrics 22, 636–669.

Dimitriadis, T., Bayer, S., 2019. A joint quantile and expected shortfall regression framework. Electronic Journal of Statistics 13, 521–547. URL: https://projecteuclid.org/journals/electronic-journal-of-statistics/volume-13/issue-1/A-joint-quantile-and-expected-shortfall-regression-framework/10.1214/19-EJS1560.full, doi:10.1214/19-EJS1560.

Dorogush, A.V., Ershov, V., Gulin, A., 2018. Catboost: gradient boosting with categorical features support. arXiv:1810.11363.

Engle, R.F., Manganelli, S., 2004. Caviar: Conditional autoregressive value at risk by regression quantiles. Journal of Business & Economic Statistics 22, 367–381. URL: http://www.jstor.org/stable/1392044.

Feng, G., He, J., Polson, N.G., 2018. Deep learning for predicting asset returns. arXiv preprint arXiv:1804.09314 .

Fuentes, F., Herrera, R., Clements, A., 2025. Tail risk dynamics of banks with score-driven extreme value models. Journal of Empirical Finance 81, 101593. URL: https://www.sciencedirect.com/science/article/pii/S0927539825000155, doi:https://doi.org/10.1016/j.jempfin.2025.101593.

Garleanu, N., Pedersen, L.H., Poteshman, A.M., 2008. Demand-based option pricing. The Review of Financial Studies 22, 4259–4299.

Garman, M.B., Kohlhagen, S.W., 1983. Foreign currency option values. Journal of International Money and Finance 2, 231–237.

Gneiting, T., 2011. Making and evaluating point forecasts. Journal of the American Statistical Association 106, 746–762. doi:10.1198/jasa.2011.r10138.

Gunnarsson, E.S., Isern, H.R., Kaloudis, A., Risstad, M., Vigdel, B., Westgaard, S., 2024. Prediction of realized volatility and implied volatility indices using ai and machine learning: A review. International Review of Financial Analysis , 103221.

Hansen, P.R., Lunde, A., Nason, J.M., 2011. The model confidence set. Econometrica 79, 453–497. URL: https://www.jstor.org/stable/41057463.

Haug, E.G., Frydenberg, S., Westgaard, S., 2010. Distribution and statistical behavior of implied volatilities. Business Valuation Review 29, 186–199.

Haug, E.G., Taleb, N.N., 2011. Option traders use (very) sophisticated heuristics, never the black–scholes–merton formula. Journal of Economic Behavior & Organization 77, 97–106.

Haugom, E., Ray, R., Ullrich, C.J., Veka, S., Westgaard, S., 2016. A parsimonious quantile regression model to forecast day-ahead value-at-risk. Finance Research Letters 16, 196–207.

Hochreiter, S., 1997. Long short-term memory. Neural computation 9, 1735–80. doi:10.1162/neco.1997.9.8.1735.

Jorion, P., 1996. Risk: Measuring the risk in value at risk. Financial Analysts Journal , 47–56.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.Y., 2017. LightGBM: A highly efficient gradient boosting decision tree. Advances in neural information processing systems 30.

Keilbar, G., Wang, W., 2022. Modelling systemic risk using neural network quantile regression. Empirical Economics 62, 93–118.

Koenker, R., Bassett, G., 1978. Regression quantiles. Econometrica 46. doi:https://doi.org/10.2307/1913643.

Koop, G., Korobilis, D., 2012. Forecasting inflation using dynamic model averaging. International Economic Review 53, 867–886.

Kupiec, P.H., et al., 1995. Techniques for verifying the accuracy of risk measurement models. volume 95. Division of Research and Statistics, Division of Monetary Affairs, Federal ….

de Lange, P.E., Risstad, M., Westgaard, S., 2022. Estimating value-at-risk using quantile regression and implied volatilities. Journal of Risk Model Validation 16, 1–24. doi:10.21314/JRMV.2021.014.

Liu, L.Y., Patton, A.J., Sheppard, K., 2015. Does anything beat 5-minute RV? A comparison of realized measures across multiple asset classes. Journal of Econometrics 187, 293–311. URL: https://www.sciencedirect.com/science/article/pii/S0304407615000329, doi:10.1016/j.jeconom.2015.02.008.

Lyócsa, Š., Plíhal, T., Vỳrost, T., 2021. Fx market volatility modelling: Can we use low-frequency data? Finance Research Letters 40, 101776.

Lyócsa, Š., Plíhal, T., Vỳrost, T., 2024. Forecasting day-ahead expected shortfall on the EUR/USD exchange rate: The (I)relevance of implied volatility. International Journal of Forecasting URL: https://www.elsevier.com/locate/ijforecast, doi:10.1016/j.ijforecast.2023.11.003.

McNeil, A.J., Frey, R., 2000. Estimation of tail-related risk measures for heteroscedastic financial time series: an extreme value approach. Journal of Empirical Finance 7, 271–300.

Nelson, D.B., 1991. Conditional heteroskedasticity in asset returns: A new approach. Econometrica 59, 347–370. URL: https://www.jstor.org/stable/2938260.

Nolde, N., Ziegel, J.F., 2017. Elicitability and backtesting: Perspectives for banking regulation. The Annals of Applied Statistics 11. URL: http://dx.doi.org/10.1214/17-AOAS1041, doi:10.1214/17-aoas1041.

Patton, A.J., Ziegel, J.F., Chen, R., 2019. Dynamic semiparametric models for expected shortfall (and value-at-risk). Journal of Econometrics 211, 388–413. URL: https://www.sciencedirect.com/science/article/pii/S030440761930048X, doi:https://doi.org/10.1016/j.jeconom.2018.10.008.

Pimentel, R., Risstad, M., Rogde, S., Stegavik, E.R., Vinje, J., Westgaard, S., Wu, C., 2025. Option pricing with deep learning: a long short-term memory approach. Data Science in Finance and Economics doi:https://doi.org/10.1007/s10203-025-00518-9.

Pimentel, R., Risstad, M., Westgaard, S., 2022. Predicting interest rate distributions using pca & quantile regression. Digital Finance 4, 291–311.

Plíhal, T., Lyócsa, Š., 2021. Modeling realized volatility of the eur/usd exchange rate: Does implied volatility really matter? International Review of Economics & Finance 71, 811–829.

Rad, H., Low, R.K.Y., Miffre, J., Faff, R., 2023. The commodity risk premium and neural networks. Journal of Empirical Finance 74, 101433.

Reiswich, D., Wystup, U., 2010. A guide to fx options quoting conventions. The Journal of Derivatives 18, 58–68.

Risstad, M., Westgaard, S., Moen, I., Pedersen, M., Utne, H.M., 2024. Conditional return quantiles, machine learning, and the implied volatility surface. Working paper .

Slim, S., Dahmene, M., Boughrara, A., 2020. How informative are variance risk premium and implied volatility for value-at-risk prediction? international evidence. The Quarterly Review of Economics and Finance 76, 22–37.

Taylor, J., 2019. Forecasting Value at Risk and Expected Shortfall Using a Semiparametric Approach Based on the Asymmetric Laplace Distribution. Journal of Business Economic Statistics 37. doi:10.1080/07350015.2017.1281815.

Vinje, J., Stegavik, E.R., Wu, C., Risstad, M., Pimentel, R., Westgaard, S., Ewald, C.O., 2025. Merged LSTM-MLP for Option Valuation. Quantitative Finance , Forthcoming.doi:https://doi.org/10.1080/14697688.2025.2493965.

Xu, X., Taylor, S.J., 1995. Conditional volatility and the informational efficiency of the PHLX currency options market. Journal of Banking & Finance 19, 803–821.

A. Regression based backtest

The Strict Expected Shortfall Regression (ESR) backtest by Bayer and Dimitriadis (2022) assesses whether the ES estimates are correctly specified by examining whether they are conditionally unbiased and appropriately scaled. It is based on the generalized ESR regression system:

$$Y_t = V_t^\top \beta + u_t^q, \quad \text{and} \quad Y_t = W_t^\top \gamma + u_t^e, \tag{A.1}$$

where Y_t is a transformation of the realized return, V_t and W_t are vectors of covariates, and u_t^q, u_t^e are residual terms. In the strict ESR backtest, we choose $V_t = W_t = (1, \hat{e}_t)$, and set up the auxiliary system:

$$Y_t = \beta_1 + \beta_2 \hat{e}_t + u_t^q \quad \text{and} \quad Y_t = \gamma_0 + \gamma_1 \hat{e}_t + u_t^e, \tag{A.2}$$

where $\hat{e}_t$ is the ES estimate at time t. Unlike traditional joint VaR-ES backtests, the strict ESR backtest requires only the ES forecast and realized returns, satisfying the strict backtesting principle formalized by Nolde and Ziegel (2017): the null holds if and only if the ES model is correctly specified. This property is particularly relevant from a regulatory perspective, as Basel III mandates separate backtesting of ES.

Formally, we test the null hypothesis:

$$H_0 := (\gamma_0, \gamma_1) = (0, 1), \tag{A.3}$$

under which the ES forecasts are conditionally unbiased ($\gamma_0 = 0$) and correctly scaled ($\gamma_1 = 1$). We employ a Wald-type test statistic, which allows for validating the precision of ES forecasts:

$$T_{\text{ESR}} = T(\hat{\gamma}_T - (0,1))^T \hat{\Omega}_\gamma^{-1} (\hat{\gamma}_T - (0,1)), \tag{A.4}$$

where $\hat{\Omega}_\gamma$ denotes a consistent estimator for the covariance of the subvector γ. Considering that the purpose is to evaluate whether the ES-estimates are correct, we adopt a two-sided ESR backtest, as opposed to a one-sided test that only evaluates if the estimates are conservative enough.

~~The null under the ESR~~ Bayer and Dimitriadis (2022) ~~is a well-specified forecast, which translates to a forecast that does neither under- nor overestimate ES. The regression system in the test is defined as follows:~~

~~$Y_t = \beta_0 + \beta_1 \hat{e}_t + u_t^\beta$~~

~~$Y_t = \gamma_0 + \gamma_1 \hat{e}_t + u_t^\gamma$~~

2005/06/

where $\hat{e}_t$ represents the ES forecasts. In this test, ES forecasts are treated as covariates in both equations for Y_t, unlike general approaches that involve joint regressions of VaR and ES, as further described in

against the alternative:

$$H_A : (\gamma_0, \gamma_1) \neq (0, 1). \tag{A.6}$$

This hypothesis testing is facilitated by a Wald-type test statistic, which allows for validating the precision of ES forecasts:

$$T_{\text{ESR}} = T(\hat{\gamma}_T - (0,1))^T \hat{\Omega}_\gamma^{-1} (\hat{\gamma}_T - (0,1)), \tag{A.7}$$

where $\hat{\Omega}_\gamma$ denotes a consistent estimator for the covariance of the subvector γ.

The strict ESR backtest uses a two-sided testing approach, advantageous in regulatory environments that require oversight to prevent from either taking on excessive risk due to underestimation or unnecessarily limiting their growth due to overestimation.

Table A1 reports p-values from the strict ESR backtest as outlined in Section A. The null hypothesis of intercept and slope coefficients equal 0 and 1, respectively, reflects correctly specified models. A non-rejection of the null, associated with high p-values, indicates that we can expect well-specified ES forecasts.

The CatBoost model's ES forecasts are generally well-specified, with some exceptions. Correct specification is evident in the lower tail for the specifications incorporating IV as well as IV and VRP. However, these models encounter challenges at the upper tail, specifically at the 0.95 and 0.975 quantiles, where the null hypothesis is rejected at a 5% significance level. Conversely, the variants incorporating IV and RR, as well as IV, RR, and VRP, generally produce well-specified forecasts across all the examined quantiles, with the exception of the 0.05 quantile. Furthermore, there is a diminishing trend as we move away from the most extreme quantiles, in terms of more frequent rejections. This pattern may indicate enhanced capabilities of predicting extreme events more accurately. Notably, all specifications , whether including or excluding the RR variable, exhibit similar overall trends. Specifically, we observe rejection of the null hypothesis for the variants utilizing RR at the 0.05 quantile, while those without RR do so at the 0.95 and 0.975 quantile.

Table A1
p-values from the Strict ES Regression Backtest

Panel A: Benchmark models						
Model	0.01 ES	0.025 ES	0.05 ES	0.95 ES	0.975 ES	0.99 ES
EGARCH-IV	0.104	0.050	0.085	0.077	0.129	0.324
EGARCH-IV-RR	0.056	0.038	0.122	0.038	0.120	0.325
EGARCH-IV-VRP	0.118	0.064	0.192	0.072	0.162	0.398
EGARCH-IV-RR-VRP	0.059	0.034	0.109	0.041	0.146	0.388
DB-IV	0.407	0.185	0.055	0.571	0.944	0.841
DB-IV-RR	0.388	0.549	0.043	0.172	0.350	0.519
DB-IV-VRP	0.004	0.195	0.135	0.561	0.617	0.701
DB-IV-RR-VRP	0.901	0.371	0.184	0.277	0.743	0.645
QR-IV	0.404	0.026	0.002	0.009	0.188	0.690
QR-IV-RR	0.016	0.013	0.002	0.012	0.289	0.673
QR-IV-VRP	0.280	0.022	0.000	0.014	0.156	0.289
QR-IV-RR-VRP	0.278	0.023	0.000	0.014	0.155	0.292
Panel B: Machine learning models						
Model	0.01 ES	0.025 ES	0.05 ES	0.95 ES	0.975 ES	0.99 ES
CatBoost-IV	0.471	0.325	0.135	0.039	0.037	0.450
CatBoost-IV-RR	0.756	0.211	0.039	0.118	0.194	0.227
CatBoost-IV-VRP	0.769	0.448	0.289	0.008	0.001	0.450
CatBoost-IV-RR-VRP	0.397	0.279	0.013	0.129	0.244	0.530
XGBoost-IV	0.485	0.428	0.329	0.004	0.017	0.678
XGBoost-IV-RR	0.869	0.475	0.068	0.037	0.188	0.282
XGBoost-IV-VRP	0.975	0.519	0.390	0.000	0.271	0.329
XGBoost-IV-RR-VRP	0.798	0.557	0.204	0.000	0.192	0.329
LightGBM-IV	0.423	0.114	0.149	0.038	0.081	0.180
LightGBM-IV-RR	0.389	0.260	0.016	0.038	0.050	0.108
LightGBM-IV-VRP	0.809	0.207	0.140	0.020	0.051	0.299
LightGBM-IV-RR-VRP	0.480	0.285	0.066	0.012	0.016	0.486
LSTM-IV	0.000	0.000	0.000	0.000	0.000	0.000
LSTM-IV-RR	0.000	0.000	0.000	0.000	0.000	0.000
LSTM-IV-VRP	0.000	0.025	0.267	0.966	0.013	0.000
LSTM-IV-RR-VRP	0.000	0.000	0.000	0.000	0.000	0.000

The values in the table are the p-values from the strict ES regression backtest under the null that the intercept is 0 and the slope coefficients are 1. A non-rejection of the null-hypothesis implies that the model is likely to produce unbiased forecasts.

For the XGBoost model, we observe rejection for the 0.95 quantile for all model specifications, in addition to the 0.975 quantile for the simplest models. The results across other quantiles display non-rejection of the null hypothesis, signifying that these models are reasonably well-specified and reliable in forecasting the ES. For XGBoost, a trend similar as for CatBoost is evident, where all variants achieve high p-values at the most extreme quantile, decreasing for less extreme quantiles.

B. Machine learning model details and hyperparameter tuning

B.1. CatBoost

The categorical boosting algorithm, CatBoost was introduced by Dorogush et al. (2018). The algorithm derives the first part of its name from categorical features. Unlike traditional gradient boosting that converts these features to numbers before training, CatBoost processes them *during* training. Another appealing attribute is its efficient strategy to mitigate overfitting while using the entire dataset for training. Specifically, the algorithm performs a random permutation of the dataset. For each example in the dataset, the average label value is derived from preceding examples in the permutation that share the identical category value. This often results in better predictive performance than XGBoost and other gradient-boosted tree algorithms (Dorogush et al., 2018). The permutation is dented by $\sigma = (\sigma_1, \dots, \sigma_n)$. To compute the transformed value for each example, the algorithm relies on:

$$\frac{\sum_{j=1}^{p-1} \left[x_{\sigma_j,k} = x_{\sigma_p,k} \right] Y_{\sigma_j} + \alpha \cdot P}{\sum_{j=1}^{p-1} \left[x_{\sigma_j,k} = x_{\sigma_p,k} \right] + \alpha} \tag{B.1}$$

where $\left[x_{\sigma_j,k} = x_{\sigma_p,k} \right]$ is an indicator function that equals 1 if the category values $x_{\sigma_j,k}$ and $x_{\sigma_p,k}$ match, and 0 otherwise. The prior value is expressed by P with a corresponding parameter $\alpha > 0$, which determines the weight of the prior. Employing a prior is a standard method to minimize noise from low-frequency categories. In a regression framework, the prior is typically calculated as the average label value in the dataset. CatBoost introduces an innovative approach for calculating leaf values, enabling multiple permutations without the risk of overfitting. CatBoost leverages oblivious trees as its base predictors, utilizing a consistent splitting criterion at each level to maintain tree balance and minimize overfitting: all features are transformed into a binary format to optimize prediction accuracy. This binary encoding method allows for efficient calculation of leaf indices, resulting in quicker and more precise model predictions. Furthermore, the entire computation process can be executed in parallel, achieving up to a threefold increase in speed, making the model exceptionally efficient (Dorogush et al., 2018). The tree structure in CatBoost is chosen through a greedy method. Features and their corresponding splits are sequentially selected for substitution in each leaf. The selection of candidates is derived from the initial split calculations and the conversion of categorical features into numerical features. The tree depth and other structural rules are determined by the initial parameters. The approach for choosing a feature-split pair for a leaf involves several steps. Initially, a list of potential candidates (feature-split pairs) is created to be considered for assignment to a leaf. Subsequently, penalty functions are calculated for each candidate, assuming they have all been allocated to the leaf. Then, the split with the least penalty is chosen. Finally, this selected value is allocated to the leaf. This process is repeated for all subsequent leaves, ensuring that the number of leaves corresponds to the tree's depth. While the literature on CatBoost primarily highlights its ability to handle categorical

features, we rely on this method for its use of oblivious trees. This enables us to evaluate the forecasting performance of a boosting model with a lower risk of overfitting compared to models using more complex tree structures.

B.2. XGBoost

The combination of decision trees and gradient boosting methods has the advantages of good training effect and not easily over-fitting. Gradient boosting is a generalization of tree boosting designed to address various issues with regular boosting, namely speed, interpretability, and, in some cases, robustness against overlapping class distributions. The XGBoost, developed by Chen and Guestrin (2016), is an ensemble model which consists of an efficient implementation of decision trees, in order to produce a combined model whose predictive performance is better than individual techniques used alone. Differently from bagging, however, boosting does not carry out bootstrap sampling, but trees are grown in a sequential basis entailing that the current generated tree exploits information from the previously generated tree. Hence, trees are no longer grown independently but sequentially dependent on construction. The additive aspect of the algorithm embodies the core principle of boosting, which iteratively adds trees to reduce the loss incrementally. This involves parameterizing each tree and adjusting these parameters to minimize the residual loss. The output of each newly added tree is then combined with the outputs of previously added trees to improve the model's overall performance. This process continues, adding a predefined number of trees until training stops, either when the loss reaches an adequate threshold or when validation loss converges (Brownlee, 2017). A tree ensemble model utilizes K additive functions to forecast the result, where T denotes the number of leaves in each tree. Each function f_k represents an independent tree structure q and leaf weights w. Regression trees assign a continuous score to each leaf, represented by w_i for the i-th leaf. Further, the decision rules specified by q in the trees determine the leaf to which any example x is assigned. The final prediction is derived by summing the scores of the relevant leaves, denoted by w. Consequently, the mathematical formulation of the additive model begins by introducing the regularized objective to be minimized as in B.2:

$$\mathcal{L}(\phi) = \sum_i l\left(\hat{y}_i, y_i\right) + \sum_k \Omega\left(f_k\right), \quad \text{where } \Omega(f) = \gamma T + \frac{1}{2}\lambda\|w\|^2 \tag{B.2}$$

where l represents the loss function and Ω denotes the regularization term, which penalizes model complexity. One of the key advantages of gradient boosting is its flexibility to accommodate various differentiable loss functions within a single boosting framework. Given that squared residuals are both the default loss function and well-suited for numerical values, we use the mean squared error (MSE) as the loss function. Incorporating the MSE loss function results in a

simplified expression, which encompasses both a quadratic and first-order residual term:

$$\mathcal{L}(\phi) = \sum_{i=1}^{n} \left(y_i - \left(\hat{y}_i^{(t-1)} + f_t\left(x_i\right) \right) \right)^2 + \sum_{i=1}^{t} \Omega\left(f_i\right) \tag{B.3}$$

In the XGBoost, several parameters need to be tuned to maximize the power of model performance and to prevent overftting problems, including the number of trees, the number of tree splits, the learning rate, the number of iterations, the maximum depth.

B.3. LightGBM

LightGBM is a gradient-boosting framework that grows trees leaf-wise (Ke et al. (2017)). In regression problems, the aim is to predict continuous values. This process starts by predicting the target variable using simple statistics like the mean or median. The model then iteratively minimises residuals to refine predictions. The unique construction of LightGBM effectively reduces losses, resulting in complex, unbalanced trees that capture detailed data set patterns. The efficiency of LightGBM in refining predictions can be represented mathematically as follows:

$$\Delta L(\theta) = -\left(\sum_{i \in I} g_i \theta + \frac{1}{2} \sum_{i \in I} h_i \theta^2 \right) \tag{B.4}$$

Here, $\Delta L(\theta)$ quantifies the change in loss, I represents the set of data points in the leaf, with g_i and h_i denoting the gradients and second-order derivatives of the loss function, respectively, for each point i.

A key strength of LightGBM is its robustness against overfitting due to L1 and L2 regularisation (**?**). These techniques give structure to the decision trees by penalising over-complex models. LightGBM's sequential tree-building approach, where each subsequent tree improves upon the errors of its predecessors, results in a robust regression model, characterised by a weighted aggregation of all the trees.

B.4. LSTM

The LSTM network provides a solution to the challenge of long-term dependencies with a structure that consists of three gates: an input gate i_t, a forget gate f_t and an output gate o_t. The subscript t represents a single time step. These gates control the flow of information used to calculate the output at each state. Figure 13 illustrates the LSTM unit structure, referred to as a memory cell, at a single time step.

The forget gate controls which parts of the cell state c_t should be discarded, using a sigmoid activation function as shown in Equation B.5. It receives as input both the previous hidden state h_{t-1} and the current input x_t. For each element in the previous cell state c_{t-1}, the gate outputs a value between 0 and 1: a value close to 0 indicates that the

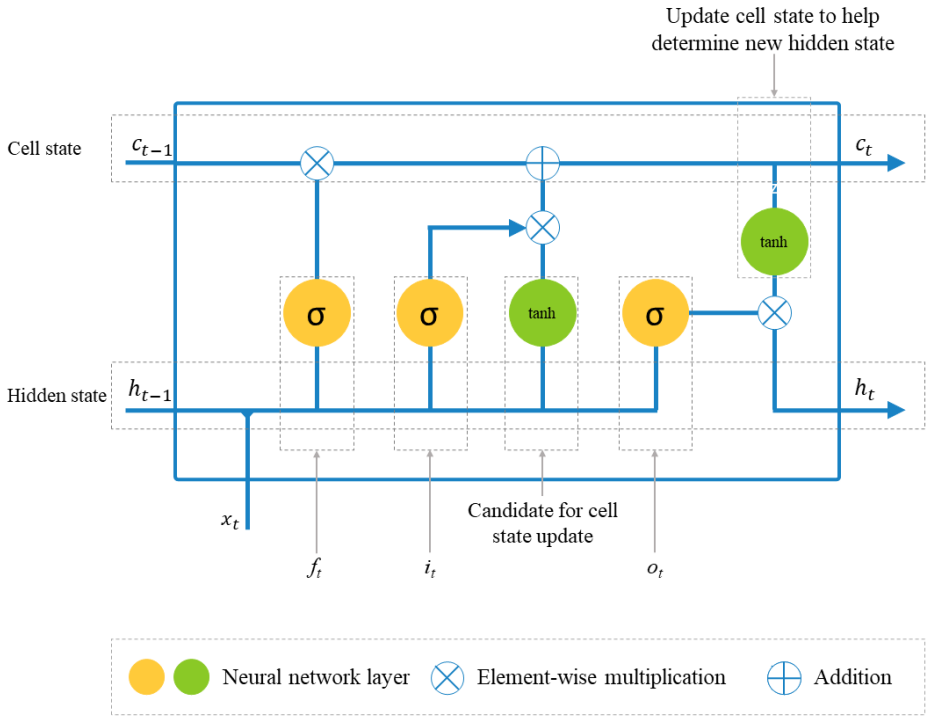

Figure 13: A single LSTM memory cell showing the forget gate, the input gate and the output gate.

information should be completely discarded, while a value close to 1 indicates that the information should be fully retained.

$$f_t = \sigma(W_{f,x}x_t + W_{f,h}h_{t-1} + b_f) \tag{B.5}$$

where σ denotes the sigmoid activation function, $W_{f,x}$ and $W_{f,h}$ denote weight matrices, and b_f denotes the bias term.

The process of determining what information to store in the cell state at the current time step can be understood in two stages. First, the input gate uses a sigmoid activation function to decide which values should be updated, as defined in Equation B.6. Next, a vector of candidate values $\tilde{c}_t$ is generated using the tanh activation function, as shown in Equation B.7. The outputs from these two stages are then combined to form the update to the cell state.

$$i_t = \sigma(W_{i,x}x_t + W_{i,h}h_{t-1} + b_i) \tag{B.6}$$

$$\tilde{c}_t = tanh(W_{\tilde{c},x} x_t + W_{\tilde{c},h} h_{t-1} + b_{\tilde{c}}) \tag{B.7}$$

where $W_{i,x}$, $W_{i,h}$, $W_{\tilde{c},x}$ and $W_{\tilde{c},h}$ denote weight matrices, and b_i and $b_{\tilde{c}}$ denote bias terms.

The update from the old cell state c_{t-1} to the new cell state c_t is given by Equation B.8.

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{B.8}$$

where $\odot$ represents element-wise multiplication.

The final output can be seen as a filtered version of the cell state c_t. First, a sigmoid layer determines which parts of the cell state should be passed on, as defined in Equation B.9. Then, the cell state is transformed by applying the tanh function, which scales its values to lie between -1 and 1. Finally, these two outputs are multiplied element-wise, ensuring that only the selected information is propagated as the output, in accordance with Equation B.10.

$$o_t = \sigma(W_{o,x} x_t + W_{o,h} h_{t-1} + b_o) \tag{B.9}$$

$$h_t = o_t \odot tanh(c_t) \tag{B.10}$$

where $W_{o,x}$ and $W_{o,h}$ denote weight matrices, and b_o denote the bias term.

B.5. Hyperparameter tuning range

Table B1 Range of hyperparameters considered for ensemble models.

Parameter	Range
Max depth[a]	3–7
Number of boosting rounds[b]	100–1000
Early stopping[c]	5–70
Learning rate[d]	0.001–0.1
L2 regularization strength[e]	1–5
Loss function	Quantile

[a] Increments of 1

[b] Increments of 50

[c] Increments of 5

[d] [0.001:0.01]: Increments of 0.001. [0.01:0.1]: Increments of 0.01.

[e] Log-scale

Table B2 Range of hyperparameters considered for the LSTM neural network.

Parameter	Range
LSTM architecture	
Layers	2:3:4
Layer units	32:64:128:256
Dropout layers	# layers – 1
Dropout rate[a]	0.1–0.5
Recurrent dropout[a]	0.0–0.5
Activation function	ReLU, tanh
Training configuration	
Epochs[b]	10–200
Batch size	16:32:64:128
Early stopping[c]	5–20
Optimizer	Adam
Initial learning rate	0.0001:0.001:0.01:0.025:0.05:0.1
Learning rate decay[d]	0.90–0.99

[a] Increments of 0.1

[b] Increments of 10

[c] Increments of 1

[d] Increments of 0.01

B.6. Alternative hyperparameter tuning

The hyperparameter tuning process in Section 3.3 involves some subjective judgment. To investigate whether this affects our main results, we implement an alternative purely data-driven hyperparameter tuning approach. We focus on CatBoost IV and IV-RR-VRP, as these are the best-performing specifications. Furthermore, we consider the left tail, as these results are considerably more heterogeneous compared to the right tail.

Under this approach, we use the check-loss function, as common for evaluating quantile estimates :

$$L_q(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^{n} \begin{cases} q\,(y_i - \hat{y}_i), & \text{if } y_i - \hat{y}_i \geq 0, \\ (1-q)\,(\hat{y}_i - y_i), & \text{if } y_i - \hat{y}_i < 0, \end{cases} \tag{B.11}$$

Using the loss-function in Equation B.11, we find a set of optimal hyperparameters for each quantile. We achieve this through a grid search over the parameter space reported in Table B1. The resulting parameters are reported in Table B3:

Table B3 Alternative hyperparameters for CatBoost

	IV			**IV-RR-VRP**		
	1.00%	2.50%	5.00%	1.00%	2.50%	5.00%
Max depth	7	7	6	7	7	4
Number of boosting rounds	789	972	701	514	514	536
Early stopping	5	5	5	5	5	5
Learning rate	0.006	0.024	0.097	0.002	0.002	0.054
L2 regularization strength	4.14	1.59	1.93	4.14	4.14	1.93

Using the retuned parameters in Table B3, we compute FZ and AL losses as per (18) and (19). This enables us to test formally whether using unique hyperparameters tuned explicitly for each quantile and specification leads to significantly lower losses.

We compute loss differentials d_t as

$$d_t = l_t^{retuned} - l_t^{original}, \tag{B.12}$$

where l_t are FZ or AL losses. The test statistic t_{NW} is

$$t_{NW} = \frac{\bar{d}}{\sqrt{\widehat{Var}_{NW}(\bar{d})}},$$

where $\bar{d}$ is the average loss differential and $\widehat{Var}_{NW}(\bar{d})$ is a Newey-West estimator that accounts for heteroskedasticity and autocorrelation at lag $T^{1/4}$.

Table B4 reports the p-values from testing the null of equal predictive ability against the alternative of the retuned hyperparameters resulting in lower average losses.

Table B4 p-values, loss differential

	IV			**IV-RR-VRP**		
	1.00%	2.50%	5.00%	1.00%	2.50%	5.00%
FZ loss (18)	0.46	0.99	0.99	0.99	0.99	0.99
AL loss (19)	0.46	0.99	0.99	0.99	0.99	0.99

Table B4 does not support rejecting the null hypothesis of equal predictive ability for the two alternative sets of hyperparameters.

C. VaR coverage tests

Kupiec test

The test statistic for the Kupiec test (Kupiec et al., 1995) is:

$$t_{Kupiec} = -2\left(n_0 \log(1-\mathrm{p}) + n_1 \log(\mathrm{p}) - \left(n_0 \log\left(1-\frac{n_1}{n}\right) + n_1 \log\left(\frac{n_1}{n}\right)\right)\right) \tag{C.1}$$

n is the number of observations, n_0 the number of observations not exceeding the threshold, n_1 the number of exceeding observations, and p the expected proportion of exceedance. The test statistic follows a chi-square distribution with one degree of freedom.

Conditional coverage test

The likelihood ratio for conditional coverage (Christoffersen, 1998) is:

$$t_{CC} = t_{UC} + t_{IND} \tag{C.2}$$

Where t_{CC} is the likelihood ratio for conditional coverage, t_{UC} is the likelihood ratio for unconditional coverage, and t_{IND} is the likelihood ratio for independence. The two components are computed as follows:

$$\begin{aligned} t_{UC} &= -2\ln\left(\frac{(1-p)^{n-n_1}\cdot p^{n_1}}{(1-\frac{n_1}{n})^{n-n_1}\cdot\left(\frac{n_1}{n}\right)^{n_1}}\right), \\ t_{IND} &= -2\ln\left(\frac{L_0}{L_1}\right). \end{aligned} \tag{C.3}$$

Where n is the total number of observations, n_1 is the number of observations exceeding VaR, and p is the expected probability of failure given by the quantile. The likelihood under the null hypothesis of independence is denoted by L_0, while L_1 represents the likelihood under the alternative hypothesis of dependence. The t_{CC} statistic follows a chi-square distribution with two degrees of freedom under the null hypothesis of correct conditional coverage.

Dynamic quantile test

The test statistic for the dynamic quantile (DQ) test (Engle and Manganelli, 2004) is given by:

$$t_{DQ} \equiv N_R^{-1}\hat{Hit}'(\hat{\beta}_{T_R})X(\hat{\beta}_{T_R})[X'(\hat{\beta}_{T_R})\cdot X(\hat{\beta}_{T_R})]^{-1}X'(\hat{\beta}_{T_R})\hat{Hit}(\hat{\beta}_{T_R})/(\theta(1-\theta)) \overset{d}{\sim} \chi^2_q \text{ as } R\to\infty. \tag{C.4}$$

Consider t_{DQ} as the out-of-sample dynamic quantile test statistic. N_R denotes the number of observations in the out-of-sample period. The vector $\hat{Hit}(\hat{\beta}_{T_R})$ represents violations, indicating when the quantile forecast is exceeded. The matrix $X(\hat{\beta}_{T_R})$ contains lagged hit terms. The quantile level is represented by θ, while χ^2_q denotes the chi-squared distribution with q degrees of freedom. R represents the index of the forecast horizon, and the condition $R \to \infty$ describes the asymptotic distribution of the test statistic.