OXFORD

A Machine Learning Approach to Volatility Forecasting*

Kim Christensen ⓘ , Mathias Siggaard and Bezirgen Veliyev

Department of Economics and Business Economics, CREATES, Aarhus University, Fuglesangs Allé 4, Aarhus V 8210, Denmark

Address correspondence to Kim Christensen, Department of Economics and Business Economics, CREATES, Aarhus University, Fuglesangs Allé 4, Aarhus V, 8210, Denmark, or e-mail: kim@econ.au.dk.

Abstract

We inspect how accurate machine learning (ML) is at forecasting realized variance of the Dow Jones Industrial Average index constituents. We compare several ML algorithms, including regularization, regression trees, and neural networks, to multiple heterogeneous autoregressive (HAR) models. ML is implemented with minimal hyperparameter tuning. In spite of this, ML is competitive and beats the HAR lineage, even when the only predictors are the daily, weekly, and monthly lags of realized variance. The forecast gains are more pronounced at longer horizons. We attribute this to higher persistence in the ML models, which helps to approximate the long memory of realized variance. ML also excels at locating incremental information about future volatility from additional predictors. Lastly, we propose an ML measure of variable importance based on accumulated local effects. This shows that while there is agreement about the most important predictors, there is disagreement on their ranking, helping to reconcile our results.

Key words: accumulated local effect, heterogeneous auto-regression, machine learning, realized variance, volatility forecasting

JEL classification: C10, C50

The unprecedented levels of volatility in financial markets during the last decades have led to increased awareness about the importance of explaining key drivers behind such movements. On the one hand, the structure of financial markets is complex, highly nonlinear, and has a low signal-to-noise ratio, which makes it nearly impossible to predict short-term asset returns (see, e.g., Gu, Kelly, and Xiu 2020; Chen, Pelger, and Zhu 2022). On the other hand, various stylized facts (e.g., the slow decay of autocorrelation in absolute returns) imply that volatility is to a large extent predictable.

With its crucial role in asset pricing, portfolio allocation, and risk management, volatility forecasting has attracted great attention. The origins of this literature go back to Engle (1982); Bollerslev (1986); Taylor (1982), who developed discrete-time GARCH and stochastic volatility models for autoregressive conditional heteroskedasticity. As argued by Corsi (2009), however, standard volatility models are not able to reproduce several salient features of financial markets. He proposes a heterogeneous autoregressive (HAR) model, which combines nonparametric realized variance measured at different frequencies with a parametric autoregressive model, in order to capture heterogeneous types of market participants and approximate long-memory of volatility. This has been an incredibly successful model, which is by now "the benchmark," to a large extent occupying the former role of the GARCH(1,1) (e.g., Hansen and Lunde 2005). However, because the HAR model is so parsimonious, the complicated structure of the underlying data renders it inadequate in some directions. Therefore, a number of extensions of the baseline HAR have been proposed (e.g., Andersen, Bollerslev, and Diebold 2007; Corsi and Renò 2012; Patton and Sheppard 2015; Bollerslev, Patton, and Quaedvlieg 2016). These are, however, still based on past returns as conditioning information, which is surprising, since a large amount of research has documented an intimate relationship between news and volatility (e.g., Schwert 1989; Engle, Ghysels, and Sohn 2013; Bollerslev, Li, and Xue 2018).

The reason that no or only few additional covariates are typically included in volatility prediction is partly explained by the observation that past volatility is a very powerful predictor of its future self. That is, after all, how this literature evolved. However, it is also motivated by the fact that traditional models, often estimated with linear regression, break down when the explanatory variables are strongly correlated, exhibit low signal-to-noise ratios, or if the underlying structure is nonlinear. As explained above, this is a trademark of financial markets, so it is unsurprising that relatively few studies conduct volatility forecasting in a data-rich environment. The recent access to large datasets therefore highlights a need for investigating more powerful tools that take these problems into account.

As emphasized by Varian (2014), machine learning (ML) techniques overcome these deficiencies. Up to now, however, only a few studies combine ML and volatility forecasting using a single approach, for example, Audrino and Knaus (2016); Audrino, Sigrist, and Ballinari (2020); Caporin and Poli (2017) investigate lasso (LA), Luong and Dokuchaev (2018) use random forests (RFs), Mittnik, Robinzonov, and Spindler (2015) apply component-wise gradient boosting (GB), Bucci (2020); Donaldson and Kamstra (1997); Hillebrand and Medeiros (2010); Fernandes, Medeiros, and Scharth (2014); and Rahimikia and Poon (2020) explore neural networks (NNs). In related work, Carr, Wu, and Zhang (2020) study the application of ML to low-frequency options data in order to forecast 30-day realized variance of the S&P 500 index.

In contrast, we conduct a comprehensive analysis of the out-of-sample performance of multiple ML techniques for volatility forecasting, that is, linear regularized models (ridge

regression [RR], LA, and elastic net [EN]), tree-based algorithms (bootstrap aggregation or bagging [BG], RF, and GB), and NNs, when a reasonably large number of additional controls appear in the information set. In such a setting, regularization is helpful to deal with the high-dimensionality of the problem and get less noisy parameter estimates. Furthermore, tree-based regression allows for estimation without imposing any distributional or functional form on the model, while NNs can regularize and capture nonlinearities. We compare these rather different ML approaches to the HAR lineage. We aim not only to do a comparison of HAR models and ML methods but also to provide evidence of how and why ML improves the accuracy of forecasting volatility. We adopt a conservative approach to implement ML in that we do minimal hyperparameter tuning to assess how good the techniques are when implemented "as is."

The findings are five-fold. First, we show superior out-of-sample forecasting with off-the-shelf implementations of ML compared with HAR. The Diebold and Mariano (1995) test for equal predictive accuracy shows that these gains are often viewed as statistically significant. Second, we document that substantial incremental information about future volatility can be extracted with ML from additional volatility predictors. In contrast, the HAR models yield only minor improvements or outright deteriorate in forecast accuracy, unless regularization is applied to account for overfitting. Third, we compute variable importance via accumulated local effect (ALE), which shows a general agreement about the set of the most dominant predictors, but disagreement on their ranking. The ALE also reveals a non-linear interaction between covariates. Fourth, we show that the forecast gains are stronger at longer horizons, which we attribute to ML techniques better capturing an underlying long-memory behavior in the realized variance. Fifth, we conduct a Value-at-Risk (VaR) application. We show that ML continues to excel at forecasting in this setting, although the differences are smaller and less statistically significant. Moreover, ML produces correct unconditional and conditional coverage.

The paper is organized as follows. Section 1 sets the theoretical foundation and introduces the various models applied in the paper. Section 2 describes the high-frequency data used in the empirical analysis. Section 3 features an out-of-sample one-day-ahead forecast comparison and conducts a series of robustness checks. In Section 4, we examine the weekly and monthly horizons. In Section 5, we perform an economic application by forecasting VaR. We conclude in Section 6. In the Appendix, we present supplemental information, including a glossary that can assist readers who are not acquainted with the jargon of ML.

1 Methodology

1.1. The Setting

We assume the log-price $X = (X_t)_{t \geq 0}$ is supported by a filtered probability space $(\Omega, (\mathcal{F}_t)_{t \geq 0}, \mathcal{F}, \mathbb{P})$. If the price is determined in an arbitrage-free frictionless market, then X is a semimartingale process, see Delbaen and Schachermayer (1994). We assume X_t can be represented as follows:

$$X_t = X_0 + \int_0^t \mu_s \mathrm{d}s + \int_0^t \sigma_s \mathrm{d}W_s + \sum_{s=1}^{N_t} J_s, \quad t \geq 0, \tag{1}$$

where X_0 is $\mathcal{F}_0$-measurable, $\mu = (\mu_t)_{t \geq 0}$ denotes the drift, $\sigma = (\sigma)_{t \geq 0}$ is the stochastic

volatility process, $W = (W_t)_{t \geq 0}$ is a standard Brownian motion, $N = (N_t)_{t \geq 0}$ is a counting process, which represents the number of jumps in X, and $J = (J_s)_{s=1,\ldots,N_t}$ is a sequence of nonzero random variables of jump sizes with corresponding jump times $\tau = (\tau_s)_{s=1,\ldots,N_t}$.

Our aim is to forecast the daily quadratic variation, which is $\mathrm{QV}_t = \int_{t-1}^{t} \sigma_s^2 \mathrm{d}s + \sum_{t-1 \leq \tau_s \leq t} J_s^2$, for $t = 1, \ldots, T$, where t is the day and T is the total number of days in the sample. The quadratic variation is not directly observable, since in practice only discretely sampled measurements of X are available. An estimator of QV is given by the realized variance:

$$\mathrm{RV}_t = \sum_{j=1}^{n} |\Delta_{t-1,j}^n X|^2, \tag{2}$$

where n is the number of intraday logarithmic returns, $\Delta_{t-1,j}^n X = X_{t-1+\frac{j}{n}} - X_{t-1+\frac{j-1}{n}}$. Then, as shown by Andersen and Bollerslev (1998) and Barndorff-Nielsen and Shephard (2002), realized variance is a consistent estimator of the quadratic variation as the sampling frequency increases, that is, $\mathrm{RV}_t \xrightarrow{\mathbb{P}} \mathrm{QV}_t$ as $n \to \infty$.

Later on, we also employ a decomposition, where realized variance is broken into its positive and negative semivariance part (e.g., Barndorff-Nielsen, Kinnebrock, and Shephard 2010):

$$\mathrm{RV}_t \equiv \mathrm{RV}_t^+ + \mathrm{RV}_t^-, \tag{3}$$

where

$$\mathrm{RV}_t^+ = \sum_{j \,:\, \Delta_{t-1,j}^n X > 0} |\Delta_{t-1,j}^n X|^2 \quad \text{and} \quad \mathrm{RV}_t^- = \sum_{j \,:\, \Delta_{t-1,j}^n X < 0} |\Delta_{t-1,j}^n X|^2. \tag{4}$$

1.2. The HAR Model and Its Extensions

The parsimonious structure of the HAR model of Corsi (2009) makes it a default choice in the literature. Therefore, the HAR model is the benchmark throughout this study, ensuring comparability with earlier work.

The HAR is defined as

$$\mathrm{RV}_t = \beta_0 + \beta_1 \mathrm{RV}_{t-1} + \beta_2 \mathrm{RV}_{t-1|t-5} + \beta_3 \mathrm{RV}_{t-1|t-22} + u_t, \tag{5}$$

where $\mathrm{RV}_{t-1|t-h} = \frac{1}{h} \sum_{i=1}^{h} \mathrm{RV}_{t-i}$. The explanatory variables are the daily, weekly, and monthly averages of lagged realized variance.

After its inception, a large number of extensions of the basic HAR has been developed. To ensure our results are robust against a broader suite of recent HAR-type models, we include several of the more popular ones as "horses" in the race. In particular, to impose a nonlinear relationship between present and future volatility and attenuate the influence of outliers, Corsi (2009) also proposed the log-version of the HAR (LogHAR):

$$\log(\mathrm{RV}_t) = \beta_0 + \beta_1 \log(\mathrm{RV}_{t-1}) + \beta_2 \log(\mathrm{RV}_{t-1|t-5}) + \beta_3 \log(\mathrm{RV}_{t-1|t-22}) + u_t. \tag{6}$$

To capture a leverage effect, Corsi and Renò (2012) suggested to include past aggregated negative returns into the HAR model, which is dubbed LevHAR:

$$\mathrm{RV}_t = \beta_0 + \beta_1 \mathrm{RV}_{t-1} + \beta_2 \mathrm{RV}_{t-1|t-5} + \beta_3 \mathrm{RV}_{t-1|t-22} + \gamma_1 r_{t-1}^- + \gamma_2 r_{t-1|t-5}^- + \gamma_3 r_{t-1|t-22}^- + u_t, \tag{7}$$

where

$$r_{t-1|t-b} = \frac{1}{b}\sum_{i=1}^{b} r_{t-i}, \tag{8}$$

and $r_{t-1|t-b}^- = \min(0, r_{t-1|t-b})$.

An alternative way to model an asymmetric impact of past positive and negative returns on future volatility is given by the semivariance HAR (SHAR) of Patton and Sheppard (2015):

$$\mathrm{RV}_t = \beta_0 + \beta_1^- \mathrm{RV}_{t-1}^- + \beta_1^+ \mathrm{RV}_{t-1}^+ + \beta_2 \mathrm{RV}_{t-1|t-5} + \beta_3 \mathrm{RV}_{t-1|t-22} + u_t. \tag{9}$$

We also add the HARQ by Bollerslev, Patton, and Quaedvlieg (2016), which corrects an inherent error-in-variables problem in the HAR that arises because realized variance is a generated regressor. The HARQ is defined as

$$\mathrm{RV}_t = \beta_0 + (\beta_1 + \beta_{1Q}\mathrm{RQ}_{t-1}^{1/2})\mathrm{RV}_{t-1} + \beta_2\mathrm{RV}_{t-1|t-5} + \beta_3\mathrm{RV}_{t-1|t-22} + u_t, \tag{10}$$

where

$$\mathrm{RQ}_t = \frac{n}{3}\sum_{j=1}^{n} |\Delta_{t-1,j}^n X|^4, \tag{11}$$

is the realized quarticity.

As readily seen, the formula for a general HAR model (except the LogHAR) is

$$\mathrm{RV}_t = \beta_0 + \beta' Z_{t-1} + u_t, \tag{12}$$

where Z_{t-1} is the information set, $\beta = (\beta_1, \ldots, \beta_J)'$ are the slope parameters, and J is the number of explanatory variables. The HAR model has $Z_{t-1} = (\mathrm{RV}_{t-1}, \mathrm{RV}_{t-1|t-5}, \mathrm{RV}_{t-1|t-22})'$, whereas the HARQ has $Z_{t-1} = (\mathrm{RV}_{t-1}, \mathrm{RQ}_{t-1}^{1/2}\mathrm{RV}_{t-1}, \mathrm{RV}_{t-1|t-5}, \mathrm{RV}_{t-1|t-22})'$.

In general, Z_{t-1} can include covariates that are not functions of past returns. We inspect such a "big data" setting. A complete list of extra explanatory variables included in this article is available in Table 1. To ensure comparability with ML, we, therefore, adopt a distributed lag-type version of each HAR model to allow for a broader selection of variables to be included as conditioning information. We denote by HAR-X the extended version of the basic HAR. However, to keep notation down the other HAR models are still denoted with their original label even in the extended setting.

While the above list is comprehensive, it is necessarily incomplete. We do not include HAR models, where the quadratic variation is split into a continuous and jump part (e.g., Andersen, Bollerslev, and Diebold 2007).

In order to estimate the coefficients of the HAR models, we employ a least squares approach with objective function:

$$(\widehat{\beta}_0, \widehat{\beta}) = \underset{\beta_0, \beta}{\operatorname{argmin}}\, \mathcal{L}(\beta_0, \beta) = \underset{\beta_0, \beta}{\operatorname{argmin}} \sum_{t \in \text{in-sample}} (\mathrm{RV}_{t+1} - \beta_0 - f(Z_t; \beta))^2, \tag{13}$$

where $f(Z_t; \beta) = \beta' Z_t$ and in-sample is defined later.

Table 1 List and summary statistics of explanatory variables

No.	Acronym	Mean		Median		Maximum		Minimum		Standard deviation		Skewness		Kurtosis	
1	RVD	20.67	[14.76,27.34]	17.26	[12.26,23.14]	176.99	[116.58,313.61]	5.32	[3.51,6.77]	12.31	[8.10,19.90]	3.34	[1.99,5.94]	26.23	[9.91,98.50]
2	RVW	21.14	[15.15,28.09]	17.77	[12.74,24.34]	116.55	[80.33,220.78]	7.40	[5.38,9.61]	11.50	[7.47,19.16]	2.85	[1.65,4.51]	16.22	[6.94,34.29]
3	RVM	21.52	[15.48,28.71]	18.04	[12.86,25.35]	88.62	[56.31,152.59]	9.26	[7.25,11.99]	10.89	[6.96,18.44]	2.55	[1.53,3.71]	12.29	[6.09,24.98]
4	IV	26.27	[19.50,37.34]	24.41	[18.35,37.07]	67.16	[49.63,86.20]	12.42	[8.44,15.73]	8.17	[5.78,11.80]	1.26	[0.47,1.70]	5.08	[3.06,7.43]
5	EA														
6	VIX	19.63		17.09		80.86		9.14		8.94		2.15		9.89	
7	EPU	101.04		83.60		719.07		3.32		68.66		2.15		11.52	
8	US3M	−0.08		0.00		76.00		−81.00		4.88		−1.46		78.07	
9	HSI	0.00		0.00		0.34		0.00		0.01		15.76		341.30	
10	M1W	0.11	[−0.11,0.35]	0.13	[−0.07,0.35]	21.98	[12.19, 51.33]	−19.22	[−34.57, −11.62]	3.08	[1.96,4.15]	0.10	[−0.59,1.34]	8.30	[4.86,30.25]
11	$VOL	0.02	[−0.03,0.08]	−1.21	[−2.19, −0.12]	210.21	[150.82, 327.68]	−162.28	[−209.44, −124.94]	33.54	[26.38, 39.68]	0.37	[0.19,0.85]	5.11	[4.31,8.47]
12	ADS	−28.99		−15.37		88.33		−379.75		69.22		−2.38		10.78	

Notes: EA is equal to one if the company makes an EA on the day of the forecast, zero otherwise. Hence, we omit the calculation of descriptive statistics for the EA variable. The square brackets contain interval-based measures of asset-specific variables. Here, the numbers are defined as the minimum and maximum value of that particular descriptive statistic when calculated individually over the cross-section of the included equities. The descriptive statistics for US3M and $VOL are for the transformed variables. ADS is reported in percent.

1.3. Regularization

When increasing the number of predictors in a setting with a low signal-to-noise ratio, the linear models at some point start to fit noise rather than relevant information, also known as overfitting. A common way to avoid overfitting and increase out-of-sample performance is to shrink the regression coefficients by imposing a penalty term.

We define the penalized loss function:

$$\tilde{\mathcal{L}}(\beta_0, \beta; \theta) = \mathcal{L}(\beta_0, \beta) + \phi(\beta; \theta),$$ (14)

where $\phi(\beta; \theta)$ is a penalty term and θ is a vector of hyperparameters, which are always determined in the validation set.[1]

In this article, we estimate three of the most widely used penalization methods, that is RR by Hoerl and Kennard (1970) and LA by Tibshirani (1996), where the penalty term is an L^2 and L^1 norm, and EN by Zou and Hastie (2005), which is a convex combination of RR and LA.

In RR, the penalty term is given by

$$\phi(\beta; \lambda) = \lambda \sum_{i=1}^{J} \beta_i^2,$$ (15)

where $\lambda \geq 0$ controls the amount of shrinkage. In the implementation, we choose a range for λ that is effectively broad enough to include both the unregularized model and the constant model as potential solutions.

If dealing with a large feature space, shrinking is not always sufficient, and subset selection is preferable. LA addresses this with a penalty term of the form:

$$\phi(\beta; \lambda) = \lambda \sum_{i=1}^{J} |\beta_i|.$$ (16)

Due to the geometry, LA often forces coefficient estimates to zero and generates sparsity and subset selection. The cost of this is that there is no closed-form solution to the problem, which makes it necessary to conduct numerical optimization.

The final penalization approach is the EN:

$$\phi(\beta; \lambda, \alpha) = \lambda \Big(\alpha \sum_{i=1}^{J} \beta_i^2 + (1-\alpha) \sum_{i=1}^{J} |\beta_i| \Big),$$ (17)

where $\alpha \in [0, 1]$.[2]

To gauge the effect of subset selection as a tool to pre-filter the data, we also include a post-lasso (P-LA) HAR, in which the penalization in Equation (16) is applied to the HAR model in a first-stage regression, and then only the variables with a nonzero slope coefficient estimate are included in an unrestricted second-stage HAR regression.

At last, following Zou (2006) we also estimate an adaptive-lasso (A-LA) HAR model. Here, the two-stage procedure of the P-LA is basically reversed. First, an unrestricted HAR

---

1   A thorough explanation of the validation procedure can be found in Appendix A.4.
2   The EN nests LA for $\alpha = 0$ and RR for $\alpha = 1$.

is estimated to get estimates of the slope parameters, say $\widehat{\beta}^{1st}$. Second, a LA is run with a modified penalty term linked to the reciprocal of the elements in $\widehat{\beta}^{1st}$:

$$\phi(\beta; \lambda) = \lambda \sum_{i=1}^{J} \widehat{w}_i |\beta_i|, \tag{18}$$

where $\widehat{w}_i = 1/|\widehat{\beta}_i^{1st}|$. Hence, a larger magnitude of the beta coefficient from the first stage leads to it being downweighted in the penalization during the second stage.[3]

1.4. Tree-Based Regression

A concern with linear models is that it is left to the researcher to impose the true association among predictors and the response variable. In contrast, a regression tree is a fully nonparametric approach, which not only allows for nonlinearity, but it also implicitly accounts for interaction effects between explanatory variables.

A tree-based regression is, as the name suggests, based on a decision tree. A tree is grown by partitioning the domain (or feature space in ML parlance) of the explanatory variables Z_t, $\text{dom}(Z_t) \subseteq \mathbb{R}^J$, into smaller and smaller rectangular subspaces as we move through the tree (an illustrative example is provided in Figure 1). This process is continued until a stopping criterion is reached. At the end of the tree, a constant prediction is assigned to all observations that fall in a given terminal node (also called a leaf).

Suppose that after constructing the tree there are M terminal nodes. They correspond to a sequence of disjoint rectangles R_m, for $m = 1, \ldots, M$, such that $R_m \subseteq \text{dom}(Z_t)$ and $\cup_{m=1}^{M} R_m = \text{dom}(Z_t)$. The tree-based regression then predicts as follows:

$$\widehat{f}(Z_t) = \sum_{m=1}^{M} \widehat{RV}_m 1_{\{Z_t \in R_m\}}, \tag{19}$$

where $\widehat{RV}_m$ is a constant.

To ensure internal consistency, we adopt the minimum sum of squares criterion for selecting $\widehat{RV}_m$, that is, $\widehat{RV}_m = \text{average}(RV_{t+1}|Z_t \in R_m)$.

In essence, tree-based regression poses a combinatorial problem. There is no closed-form solution to find the best order in which to sort and split variables, and it may be too time-consuming to search over all possible choices. Hence, the need for a short-sighted, greedy algorithm arises. A popular approach to this problem is to start at the top node (called the root) and independently optimize nearby nodes in a stepwise fashion, known as the Classification And Regression Tree (CART). The CART, however, often yields inferior out-of-sample performance due to high variance from overfitting sample noise.

Breiman (1996) proposed BG to refine the decision tree. In BG, we grow several trees by resampling the original data. We subsequently predict as follows:

$$\widehat{f}(Z_t) = \frac{1}{B} \sum_{b=1}^{B} \widehat{f}^b(Z_t), \tag{20}$$

where $\widehat{f}^b(Z_t)$ is the tree-based prediction from the bth bootstrap sample.

---

3   All variables are standardized prior to estimation to make the slope parameters scale invariant.

$$RV_{t-1} \leq q_{RV_{t-1}}^{0.50} = 26.10 \ [3,388]$$

$$RV_{t-1} \leq q_{RV_{t-1}}^{0.25} = 18.44 \ [1,694] \qquad RV_{t-1|t-5} \leq q_{RV_{t-1|t-5}}^{0.50} = 27.77 \ [1,694]$$

$$\widehat{RV}_1 = 17.58 \ [847] \qquad \widehat{RV}_2 = 25.22 \ [847] \qquad \widehat{RV}_3 = 30.15 \ [242] \qquad \widehat{RV}_4 = 45.63 \ [1,452]$$
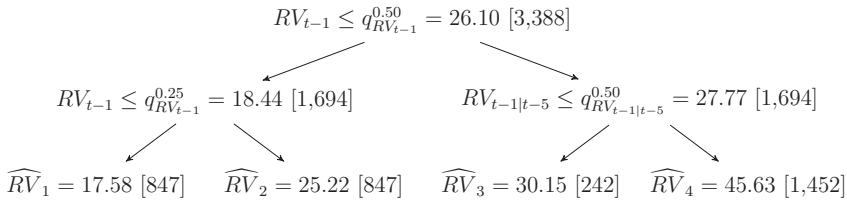
Figure 1 Illustration of a regression tree.

Notes: The figure shows an example of a possible regression tree constructed with the actual values of RV_{t-1}, $RV_{t-1|t-5}$, and $RV_{t-1|t-22}$ from the training and validation set of Apple (AAPL) high-frequency data in our empirical application. The variables are expressed as annualized standard deviation in percent for ease of interpretation. We arbitrarily set a stopping criteria to a maximum number of terminal nodes $M = 4$ and select the split points conveniently as the first and second quartile ($q^{0.25}$ and $q^{0.50}$) of the associated variable for the purpose of the illustration. The numbers in square brackets are the total data points retained at each node, as we move through the tree, while $\widehat{RV}_m$, $m = 1, \ldots, 4$, corresponds to the one-day-ahead prediction of RV_t at each terminal node. Note the uneven distribution of data in the right-hand side of the tree is caused by using the overall median of $RV_{t-1|t-5}$ as a split point.

The serial correlation in volatility suggests a block bootstrap may be appropriate. However, in our experience an i.i.d. bootstrap does not impair the recursive segmentation of the input space compared with more refined resampling.[4]

The benefit of BG compared with CART is related to the reduced correlation between trees. Therefore, an alternative approach building on BG is RF, see Breiman (2001). Here, attention is restricted to a random subset of input features before finding the best split point. In this way, a RF is able to de-correlate the trees further. In particular, B trees $\{f(Z_t^b, \theta^b)\}_{b=1}^B$ are grown, where θ^b summarizes the bth RF tree in terms of split variables, split points, and values at the terminal nodes. The prediction is then

$$\widehat{f}(Z_t) = \frac{1}{B} \sum_{b=1}^{B} \widehat{f}^b(Z_t, \theta^b). \tag{21}$$

To reduce the complexity of the RF and BG, we set all tuning parameters equal to their default values from the original Fortran code by Breiman and Cutler (2004). This sidesteps validation and facilitates reproduction of our findings.

Another tree-based approach is GB by Friedman (2001).[5] GB produces a sequential model based on weak learners, where each of B trees is grown on information from the previous one. First, $\widehat{f}^0(Z_t)$ is initialized as a constant determined by

---

4   As a verification, we applied a block bootstrap with B random samples from a given dataset $\mathcal{M}$, resulting in a sequence $\mathcal{M}^b$, for $b = 1, \ldots, B$, where the block length l and the number of blocks in each bootstrap sample k was chosen such that $T = kl$. l was found in the validation set. We estimated a regression tree for each bootstrap and computed the prediction function as $\widehat{f}(Z_t) = \frac{1}{B} \sum_{b=1}^{B} \widehat{f}^b(Z_t)$. There was no discernible change in the results compared with the i.i.d. bootstrap.

5   He extends the original boosting algorithm from Schapire (1990) and Freund (1995).

$\widehat{f}^0(Z_t) = \mathrm{argmin}_{\beta_0} \sum_{t \in \mathrm{in-sample}} (\mathrm{RV}_t - \beta_0)^2$. With the mean-squared error (MSE) loss function, this constant is the sample average. Secondly, the negative gradient of the loss function with respect to the prediction is calculated. With the MSE loss function, this corresponds to the residuals. A shallow tree is then fitted to the residuals, yielding a set of terminal nodes $R_{jb}, j = 1, \ldots, J_b$, where j denotes the leaf and b the tree. This is followed by choosing a gradient descent size as $\rho^{jb} = \mathrm{argmin}_{\beta_0} \sum_{Z_t \in R_{jb}} (\mathrm{RV}_t - \beta_0 - \widehat{f}^{b-1}(Z_t))^2$, and in the last step $\widehat{f}^b(Z_t)$ is updated iteratively as follows:

$$\widehat{f}^b(Z_t) = \widehat{f}^{b-1}(Z_t) + \nu \sum_{j=1}^{J_b} \rho^{jb} 1_{\{Z_t \in R_{jb}\}}, \tag{22}$$

for $b = 1, \ldots, B$, where ν is a learning rate and $\widehat{f}(Z_t) \equiv \widehat{f}^B(Z_t)$ the final prediction.

As explained by de Prado (2018), GB attempts to address an underfitting problem, whereas RF deals with overfitting (generally perceived to be a greater evil in finance). As a consequence, GB places substantially more weight on misclassification and is therefore susceptible to outliers. With the typical heavy tails in the volatility distribution, we expect this to lead to poor results for GB in our framework.

1.5. Neural Network

To ensure a complete investigation of ML space, at last we apply the artificial NN. With their highly flexible and nonlinear structure, NNs show astonishing performance in handling complex problems. However, with a broad set of hyperparameters the replicability of the NN is frequently questioned. Therefore, we construct the NN as streamlined as possible and set the majority of the hyperparameters to standard suggestions from the literature. This is presented in Appendix A.4.

We briefly outline the mechanics of a so-called feed-forward network, see Figure 2 for an example. A more comprehensive overview is available in, for example, Goodfellow, Bengio, and Courville (2016). A NN is constructed from an input layer, hidden layers, and an output layer. Suppose the total number of such layers is L. In the first layer, the NN receives an input Z_t. The data are transformed through one or more hidden layers using an activation function g.

The general equation for the lth layer is denoted as

$$a_t^{\theta_{l+1}, b_{l+1}} = g_l \left(\sum_{j=1}^{N_l} \theta_j^{(l)} a_t^{\theta_l, b_l} + b^{(l)} \right), \quad 1 \leq l \leq L, \tag{23}$$

where g_l is the activation function, $\theta^{(l)}$ is the weight matrix, $b^{(l)}$ is the error serving as an activation threshold for the neurons in the next layer, N_l denotes the number of hidden neurons, and $a_t^{\theta_{l+1}, b_{l+1}}$ is the prediction.[6]

The output layer constructs a forecast

$$\widehat{f}(Z_t) = (g_L \circ \cdots \circ g_1)(Z_t). \tag{24}$$

As seen, a NN applies a series of functional transformations to construct the forecast. In principle, a single hidden layer with a large enough number of neurons (and appropriate

---

6   By definition, the prediction from the input layer is $a_t^{\theta_2, b_2} = g_1(\sum_{j=1}^{N_1} \theta_j^{(1)} Z_t + b^{(1)})$.
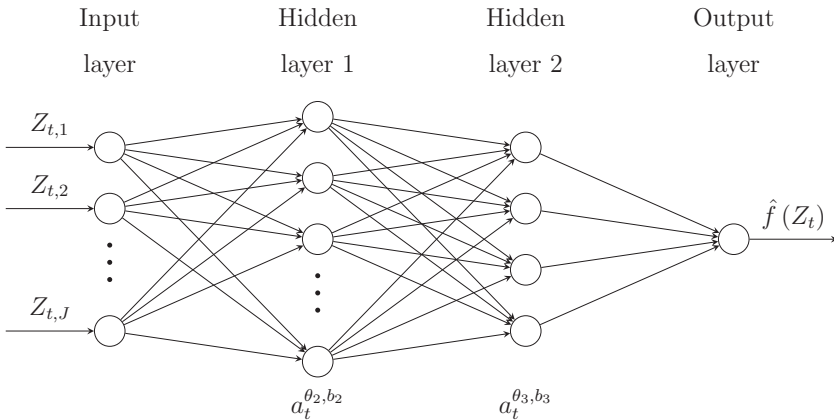
Figure 2 Feed-forward NN.

Notes: We illustrate a two-layered feed-forward NN with an input layer, a pair of hidden layers, and the output layer. The input layer receives the raw features Z_t with J explanatory variables. Each hidden layer computes an affine transformation and applies an element-wise activation function. The output layer returns the prediction. The arrows denote the direction of the information flow throughout the network during forward propagation.

activation function) is sufficient to approximate any continuous function. This follows from the Universal Approximation Theorem of Cybenko (1989). In practice, however, it is often convenient and computationally more efficient to add extra hidden layers than to arbitrarily increase the number of neurons in a layer. Hence, the optimal architecture of a NN depends on the problem and is determined through tuning.

To allow inspection of the inner workings of a NN, we construct four models with an architecture inspired by the geometric pyramid (see, e.g., Masters 1993; Gu, Kelly, and Xiu 2020). The first, denoted NN_1, has a shallow structure with a single hidden layer and 2 neurons. NN_2 is two-layered with 4 and 2 neurons, NN_3 has three layers with 8, 4, and 2 neurons, and NN_4 has four layers with 16, 8, 4, and 2 neurons.

We apply the Leaky Rectified Linear Unit (L-ReLU) by Maas, Hannun, and Ng (2013) as activation function for all layers, as it is able to infer when the activation is zero via gradient-based methods, in contrast to the standard ReLU.[7] The L-ReLU is defined as

$$\text{L-ReLU}(x) = \begin{cases} cx, & \text{if } x < 0, \\ x, & \text{otherwise,} \end{cases} \qquad (25)$$

where $c \geq 0$.[8]

The weight parameters are chosen to minimize the squared error loss function.[9] In lack of a closed-form solution, we adopt the Adaptive Moment Estimation (Adam) by Kingma

---

7　Other common activation functions include Sigmoid, TanH, or ReLU. Switching to one of these has no material impact on our empirical results.

8　In this article, $x = \sum_{j=1}^{N_l} \theta_j^{(l)} a_t^{\theta_l, b_l} + b^{(l)}$. We follow the majority of the literature by setting $c = 0.01$.

9　Alternatively, a penalized L^2 objective function can be applied. We found no additional improvement when combining L^2 and drop-out.

and Ba (2014), where the optimizer updates the weights $(\widehat{\theta}, \widehat{b})$ as a combination of Momentum and Root Mean Square propagation (RMSprop).[10] We further predetermine the learning rate at 0.001, see Kingma and Ba (2014).

Regularization of a NN is crucial to avoid overfitting. There is a large literature proposing different regularization techniques. In this study, we employ four of the most commonly used, namely learning rate shrinkage, drop-out, early stopping, and ensembles. Ensembling is a model averaging technique, in which numerous NNs are fitted, and the prediction is constructed from a subset of the best performing models from the validation set. The idea is to reduce the variation in the forecast caused by the random initial weights assigned to a NN by repeating the experiment several times, much as an ensemble average. We investigate ensembles of 1—corresponding to no averaging—and 10 (out of 100), which we denote NN_1^1 and NN_1^{10} for the single hidden layer NN, and so forth. In Appendices A.4 and A.5, we explain the regularization techniques and choice of hyperparameters in more detail.

1.6. Forecast Comparison

As the MSE laid the foundation for the optimal structure of the ML algorithms, a logical continuation is to employ it also as an out-of-sample evaluation measure:

$$\mathcal{L}(\widehat{f}(Z_t)) = \sum_{t \in \text{out-of-sample}} (RV_{t+1} - \widehat{f}(Z_t))^2, \tag{26}$$

where $\widehat{f}(Z_t)$ is the forecast.[11]

If a model predicts volatility to be negative, we replace the forecast with the minimum in-sample realized variance $\min_{t \in \text{in-sample}} RV_t$. We also adopt the insanity filter from Bollerslev, Patton, and Quaedvlieg (2016) for the HARQ.

To measure statistical significance, we compute a pairwise Diebold and Mariano (1995) test for equal predictive accuracy with a one-sided alternative that the comparison model beats the benchmark model, as explained later. We also construct a Model Confidence Set (MCS) of Hansen, Lunde, and Nason (2011). It defines a collection of models containing the "best" one with a given level of confidence. Inferior forecasting models are removed via an elimination rule.

2 Data Description

The empirical investigation is based on high-frequency data from 29 of the 30 Dow Jones Industrial Average (DJIA) constituents prior to the recomposition on August 31, 2020. The complete list of ticker symbols is AAPL, AXP, BA, CAT, CSCO, CVX, DIS, DOW, GE, GS, HD, IBM, INTC, JNJ, JPM, KO, MCD, MMM, MRK, MSFT, NKE, PFE, PG, RTX, TRV,

---

10   To limit the number of hyperparameters and reduce the complexity of the optimization, we select exponential decay rates in line with Kingma and Ba (2014).

11   The LogHAR produces a forecast $\widehat{f}(Z_t)$ of log-realized variance, but we require a forecast of realized variance $\exp(\widehat{f}(Z_t))$. This is a nonlinear transformation, so the forecast of realized variance is biased by Jensen's inequality, even if the forecast of log-realized variance is not. We bias-correct as follows: $E[\exp(\widehat{f}(Z_t))] = \exp\left(E[\widehat{f}(Z_t)] + 0.5\mathrm{var}[\widehat{f}(Z_t)]\right)$, where $\mathrm{var}(\widehat{f}(Z_t))$ is the variance of the residuals in the training and validation set. This is suitable when the distribution of log-realized variance is Gaussian, which is approximately true in practice, see, for example, Andersen et al. (2001) and Christensen, Thyrsgaard, and Veliyev (2019).

UNH, VZ, WMT, and XOM.[12] Visa (V) is excluded due to limited data availability, as its shares were only listed for public trading on NYSE on March 19, 2008. We construct a continuous stock price record for each company on a sample from January 29, 2001 to December 31, 2017, or $T = 4,257$ trading days. The sample includes several noteworthy events, such as the financial crisis, the European sovereign debt crisis, several rounds of negotiations about the limit of the U.S. debt ceiling, and the flash crashes of May 6, 2010 and August 24, 2015.

There has been several corporate actions changing the ticker symbol history for some of the current members of DJIA. We include high-frequency information from the main predecessor prior to such events. In particular, we employ (i) Chevron prior to its merger with Texaco on October 9, 2001 to forge the company ChevronTexaco (CVX), (ii) St. Paul Travelers Companies (STA) prior to its name change to Travelers Inc. (TRV) on February 27, 2007, (iii) United Technologies (UTX) prior to its replacement by Raytheon Technologies (RTX) on April 6, 2020 (Raytheon is the fusion of United Technologies and Raytheon Company, which merged as of April 3, 2020), (iv) Walgreens (WAG) prior to forming Walgreens Boots Alliance by purchasing a majority stake in Alliance Boots on December 31, 2014, and (v) DowDuPont (DWDP) before the spin-off of Dow Chemical Company (DOW) on April 2, 2019. DowDupont itself was the result of a previous merger between Dow Chemical Company and DuPont (DD) on September 1, 2017, and here we take DuPont as predecessor.

The data are extracted from the NYSE Trade and Quote database. Prior to our analysis, the data were pre-processed and filtered for outliers using standard algorithms, see Barndorff-Nielsen et al. (2009) and Christensen, Oomen, and Podolskij (2014). Based on the cleaned transaction price record, we construct a 5-min log-return series and compute for each asset a time series of daily realized variance RV_t with $n = 78$, for $t = 1, \ldots, 4,257$. The goal is to forecast realized variance.

We split the data into a training, validation, and test set. The primary analysis is restricted to a training set of 70% of the dataset (or 2,964 days), a validation set of 10% (or 424 days), and a test set of 20% (or 847 days).[13] The latter are reserved for out-of-sample evaluation. We conduct a robustness check, where the training set is modified to 1,000 and 2,000 days (with a validation set fixed at 200 days). The results, which are broadly in line with our reported findings, are available in Appendix A.1.

In the non-regularized HAR models, we merge the training and validation set and employ time-varying parameters by constructing a rolling window forecast. This approach is also adopted for BG and RF to avoid any hyperparameter tuning, which results in off-the-shelf implementation of these methods. However, tuning is almost unavoidable for RR, LA, EN, and GB. In keeping with the spirit of our paper, a rolling scheme without

---

12   We also ran the forecast comparison for the composite DJIA index and the SPY. The latter is a liquid ETF tracking the S&P 500. The outcome of this analysis confirms the findings reported for individual equities, suggesting our conclusions hold more broadly on a market-wide level. Moreover, we attempted to forecast volatility of the DJIA index based on cross-sectional information of the volatilities from the individual member firms. However, we did not find any improvement in forecast accuracy, once we controlled for past DJIA index volatilities.

13   A month worth of high-frequency data (or 22 days) are "lost" due to the lagging.

concatenation of the training and validation set is conducted. An in-depth explanation of our tuning of hyperparameters is available in Appendix A.4. The vast majority of the hyperparameters are set equal to their default values proposed in the original research papers, often based on very different contexts. At last, we employ a fixed window estimation for the NNs, where the weights are only found once in the initial validation sample and not rolled forward in the out-of-sample window. This implementation is strictly advantageous to the other competing models, which are allowed to adapt to new information. It may be possible to further improve the performance of the NNs with a rolling window approach, however this requires a very large computer capacity, which is outside our budget.

To study the effect of tagging on additional explanatory variables, we construct two datasets $\mathcal{M}_{HAR}$ and $\mathcal{M}_{ALL}$. $\mathcal{M}_{HAR}$ contains the daily, weekly, and monthly lag of realized variance (RVD, RVW, and RVM).[14] This ensures one-to-one comparison between the HAR models and ML algorithms. $\mathcal{M}_{ALL}$ extends $\mathcal{M}_{HAR}$ by including additional variables examined in previous research. We select a total of nine other key variables that are arguably important predictors of future volatility. In particular, we extract four firm characteristics: Model-free implied volatility (IV),[15,16] an indicator for earnings announcements (EAs), 1-week momentum (M1W), and dollar trading volume ($VOL). We include five macroeconomic indicators: The CBOE volatility (VIX) index, the Hang Seng stock index daily squared log-return (HSI), the Aruoba, Diebold, and Scotti (2009) business conditions (ADS) index, the US 3-month T-bill rate (US3M), and the economic policy uncertainty (EPU) index from Baker, Bloom, and Nicholas (2016).[17] We first difference US3M to account for possible nonstationarity and we replace $VOL by first-differencing its logtransform. Moreover, in the LogHAR we also log-transform VIX and IV.

A full list of explanatory variables and their acronyms is presented in Table 1, along with some standard descriptive statistics. Prior to estimation, we standardize the input data with the sample mean and sample variance from the training set to render the estimation scale invariant and improve the numerical optimization procedure for RR, LA, EN, and the NN.

---

14   $\mathcal{M}_{HAR}$ also includes variables that are employed in the various extensions of the HAR model. However, these variables are only passed to the concrete model (i.e., the square-root of realized quarticity appears as an interaction term with realized variance in HARQ, but not in the other models).

15   The data were downloaded from the OptionMetrics' IvyDB database, which estimates the IV for each traded option contract. We followed the data cleaning suggested by Lei, Wang, and Yan (2020) to deal with erroneous data points.

16   To account for the correlation between IV and realized variance, we also estimated an auxiliary model: $IV_t = \beta_0 + \beta_1 RV_t + u_t$ and employed the residuals from the fitted regression in place of IV. The results were roughly unchanged.

17   We also constructed an intermediate dataset via subset selection. In particular, we fitted a separate least squares regression for each combination of the feature space of $\mathcal{M}_{ALL}$. The best model was then selected with a BIC criterion. It contains the daily and weekly lag of realized variance, IV, M1W, and the EA indicator, that is, a total of five variables. The conclusions from the forecast comparison on this dataset is consistent with what we report below. Interestingly, if the evaluation criterion is changed from BIC to cross-validation error, the subset selection includes only the standard three HAR variables.

Table 2 One-day-ahead relative MSE and Diebold–Mariano test for dataset $\mathcal{M}_{\mathrm{HAR}}$

	HAR	HAR-X	LogHAR	LevHAR	SHAR	HARQ	RR	LA	EN	A-LA	P-LA	BG	RF	GB	NN_1^1	NN_1^{10}	NN_2^1	NN_2^{10}	NN_3^1	NN_3^{10}	NN_4^1	NN_4^{10}
HAR	–	1.000	0.995	1.073	1.009	1.059	1.000	1.003	0.999	1.007	1.005	1.147	1.020	1.054	0.980	*0.969*	*0.966*	*0.958*	*0.955*	**0.954**	0.984	0.990
HAR-X	1.000	–	0.995	1.073	1.009	1.059	1.000	1.003	0.999	1.007	1.005	1.147	1.020	1.054	0.980	*0.969*	*0.966*	*0.958*	*0.955*	**0.954**	0.984	0.990
LogHAR	1.008	1.008	–	1.076	1.017	1.063	1.007	1.010	1.006	1.016	1.013	1.151	1.025	1.059	0.987	0.976	0.972	0.964	0.962	*0.960*	0.990	0.995
LevHAR	0.947	0.947	0.938	–	0.954	0.988	0.946	0.949	0.945	0.954	0.952	1.077	0.961	0.992	0.925	0.916	0.912	*0.905*	0.902	*0.900*	0.928	0.933
SHAR	0.994	0.994	0.989	1.066	–	1.055	0.994	0.997	0.993	1.002	0.999	1.140	1.014	1.048	0.975	*0.963*	0.961	0.952	*0.950*	0.948	0.978	0.984
HARQ	0.968	0.968	0.959	1.025	0.978	–	0.966	0.970	0.965	0.975	0.973	1.104	0.982	1.015	0.944	0.935	0.932	0.924	0.921	0.920	0.948	0.953
RR	1.001	1.001	0.995	1.073	1.010	1.058	–	1.003	0.999	1.008	1.006	1.147	1.020	1.054	0.981	*0.969*	*0.966*	**0.958**	0.956	**0.954**	0.984	0.990
LA	_*0.998*_	_*0.998*_	0.992	1.070	1.007	1.056	0.997	–	0.996	1.005	***1.003***	1.143	1.017	1.051	0.978	**0.966**	0.964	**0.955**	**0.953**	**0.951**	0.981	0.987
EN	1.002	1.002	0.996	1.073	1.011	1.059	1.001	1.004	–	1.009	1.007	1.147	1.020	1.055	0.981	*0.970*	0.967	**0.959**	0.956	**0.955**	0.985	0.990
A-LA	**0.993**	**0.993**	0.988	1.065	1.002	1.052	0.992	0.995	0.992	–	0.998	1.139	1.012	1.047	0.973	*0.962*	0.959	**0.951**	0.949	**0.947**	0.977	0.982
P-LA	**0.995**	**0.995**	0.990	1.067	1.003	1.054	0.995	0.998	0.994	1.003	–	1.141	1.015	1.049	0.975	*0.964*	0.961	**0.953**	0.951	**0.949**	0.979	0.985
BG	*0.891*	*0.891*	0.882	0.947	0.898	0.936	*0.889*	*0.892*	*0.888*	0.897	0.896	–	*0.901*	0.930	0.871	**0.862**	0.859	**0.851**	0.849	**0.847**	0.874	0.879
RF	0.986	0.986	0.978	1.052	0.994	1.037	0.985	0.987	0.984	0.993	0.991	1.121	–	1.034	*0.964*	**0.954**	0.951	**0.943**	0.940	**0.938**	0.968	0.973
GB	0.958	0.958	0.949	1.020	0.966	1.008	0.957	0.960	0.956	0.965	0.963	1.088	0.972	–	*0.938*	**0.927**	0.923	0.916	0.913	0.912	0.940	0.945
NN_1^1	1.022	1.022	1.016	1.093	1.032	1.075	1.021	1.025	1.021	1.030	1.028	1.170	1.040	1.076	–	0.989	0.986	0.978	0.976	0.974	1.004	1.010
NN_1^{10}	1.033	1.033	1.027	1.106	1.042	1.091	1.032	1.036	1.032	1.041	1.039	1.184	1.053	1.087	1.012	–	0.997	0.989	0.986	0.984	1.015	1.021
NN_2^1	1.037	1.037	1.030	1.109	1.046	1.094	1.036	1.039	1.035	1.045	1.042	1.187	1.056	1.091	1.015	1.004	–	0.992	0.989	0.987	1.018	1.024
NN_2^{10}	1.045	1.045	1.038	1.118	1.054	1.103	1.044	1.047	1.043	1.053	1.051	1.197	1.064	1.100	1.023	1.012	1.008	–	0.997	0.996	1.027	1.032
NN_3^1	1.048	1.048	1.042	1.122	1.058	1.105	1.047	1.051	1.047	1.056	1.054	1.200	1.067	1.103	1.027	1.015	1.012	1.003	–	0.999	1.030	1.036
NN_3^{10}	1.050	1.050	1.043	1.123	1.059	1.107	1.049	1.052	1.048	1.058	1.056	1.202	1.069	1.105	1.028	1.016	1.013	1.005	1.002	–	1.031	1.037
NN_4^1	1.019	1.019	1.012	1.089	1.029	1.074	1.018	1.021	1.018	1.027	1.025	1.167	1.037	1.072	0.998	0.987	0.983	**0.975**	0.972	_0.970_	–	1.006
NN_4^{10}	1.014	1.014	1.007	1.083	1.024	1.067	1.013	1.016	1.012	1.022	1.020	1.160	1.032	1.066	0.993	*0.981*	0.978	**0.970**	0.967	_0.965_	0.995	–

Notes: We report the out-of-sample realized variance forecast MSE of each model in the selected column relative to the benchmark in the selected row. Each number is a cross-sectional average of such pairwise relative MSEs for each stock. The formatting is as follows: *number* (**number**) [_**number**_] denotes whether the Diebold–Mariano test of equal predictive accuracy is rejected more than 50% of the time at the 10% (5%) [1%] level of significance across individual tests for each asset. The hypothesis being tested is $H_0 : \mathrm{MSE}_i = \mathrm{MSE}_j$ against a one-sided alternative $H_1 : \mathrm{MSE}_i > \mathrm{MSE}_j$, where model i is the label of the selected row, whereas model j is the label of the selected column.

3 Empirical Results

3.1. One-Day-Ahead Forecasting

In Table 2, reading by column we report the out-of-sample MSE of each forecasting model relative to the benchmark in the selected row for the dataset $\mathcal{M}_{HAR}$. The ratio is a cross-sectional average of the relative one-day-ahead realized variance forecast MSEs for each stock. Hence, the first row contains the out-of-sample MSE of every model relative to the basic HAR, averaged over stocks. In addition, for each stock and pairwise comparison, we compute the Diebold–Mariano test. The number formatting shows whether the null hypothesis of equal predictive accuracy for a fixed pairwise comparison was rejected more than half of the times across stocks at various levels of significance. Note that in $\mathcal{M}_{HAR}$ HAR-X is restricted access to lagged realized variance and is therefore identical to HAR.

Several interesting findings emerge. First, the basic HAR is superior to its offspring. Compared with the extant literature, this is surprising. However, the results naturally depend on the split between training, validation, and test set. With a conventional shorter training set of 1,000 days, the extended HAR models outperform HAR by some distance, as reported in Appendix A.1. A possible reconciliation of this is that with a shorter in-sample size, the financial crisis appears in the out-of-sample set, which has a huge impact on the relative performance of the models. For example, the quarticity correction in the HARQ is arguably more crucial in high volatility environments. Second, when the covariate list is restricted to $\mathcal{M}_{HAR}$ the ML results are also mixed. This is not surprising. The lagged values of realized variance are strong determinants of future volatility, so there is not much room for improvement. In alignment with this interpretation, regularization is at par with HAR sporting an aggregated MSE ratio at or above one. So, as expected, there is no impending need to penalize the HAR model in $\mathcal{M}_{HAR}$. The regression trees are also inferior to HAR, especially BG, whereas the RF is about equal in performance. In contrast, the cross-sectional relative MSEs for the NNs are consistently below one by roughly 5% on average, and the Diebold–Mariano test is frequently rejected at the 10% or 5% significance level, when a NN is evaluated against the HAR as a benchmark. This shows that for short-horizon forecasts of next-day volatility, there are minor refinements associated with allowing for nonlinearities in the dynamic. What is perhaps the most striking outcome of Table 2 is that even a shallow NN beats the HAR. This can possibly be explained by the fact that a limited complexity of a NN allows the benefits of early stopping and dropout to be emphasized.

Turning next to Table 3, we show the corresponding results for $\mathcal{M}_{ALL}$. Here, the supplemental exogenous predictors from Table 1 appear in the information set. The addition of these variables causes a huge impact on the pairwise comparisons. This forcefully demonstrates the complexity and low signal-to-noise ratio in the underlying financial data, as inferred from the MSE ratio of the HAR-X and HAR. As gauged by comparing Tables 2 and 3, there is but a minuscule decrease in MSE of 3.4% moving from $\mathcal{M}_{HAR}$ to $\mathcal{M}_{ALL}$. Furthermore, the LevHAR, SHAR, and HARQ fare worse than HAR, even though their information set is larger. This supports our argument in Section 2.3 that lack of regularization leave the linear HAR models exposed to in-sample overfit. As such, RR is expected to alleviate this problem. This is also confirmed in Table 3. The ℓ_2 regularization enforced by RR reduces the average MSE of 8.1% overall. The effect of allowing for subset selection can be uncovered by contrasting LA with RR. The latter is marginally better than the

Table 3 One-day-ahead relative MSE and Diebold–Mariano test for dataset $\mathcal{M}_{\text{ALL}}$

	HAR	HAR-X	LogHAR	LevHAR	SHAR	HARQ	RR	LA	EN	A-LA	P-LA	BG	RF	GB	NN$_1^1$	NN$_1^{10}$	NN$_2^1 Z_2^1$	NN$_2^{10}$	NN$_3^1$	NN$_3^{10}$	NN$_4^1$	NN$_4^{10}$
HAR	–	0.966	0.901	1.003	1.080	1.289	0.919	0.936	0.916	0.957	0.987	0.961	0.901	0.962	0.902	0.889	0.893	0.885	0.910	0.898	0.929	0.944
HAR-X	1.045	–	0.938	1.034	1.128	1.298	0.954	0.972	0.952	0.994	1.020	1.000	0.940	0.998	0.941	0.928	0.932	0.923	0.944	0.935	0.966	0.980
LogHAR	1.113	1.070	–	1.109	1.201	1.413	1.019	1.039	1.017	1.062	1.093	1.067	1.001	1.066	1.002	0.988	0.992	0.983	1.009	0.997	1.031	1.047
LevHAR	1.018	0.971	0.912	–	1.096	1.246	0.928	0.946	0.927	0.966	0.991	0.971	0.915	0.970	0.916	0.904	0.907	0.899	0.918	0.910	0.939	0.953
SHAR	0.942	0.909	0.848	0.942	–	1.219	0.865	0.881	0.863	0.900	0.928	0.903	0.847	0.904	0.851	0.837	0.840	0.832	0.857	0.845	0.874	0.887
HARQ	0.882	0.832	0.788	0.852	0.950	–	0.799	0.813	0.799	0.831	0.847	0.837	0.792	0.832	0.796	0.785	0.788	0.781	0.791	0.787	0.809	0.818
RR	1.093	1.050	0.983	1.087	1.181	1.375	–	1.018	0.998	1.042	1.070	1.049	0.984	1.046	0.985	0.971	0.975	0.966	0.989	0.979	1.012	1.027
LA	1.076	1.033	0.967	1.070	1.161	1.352	0.984	–	0.981	1.024	1.052	1.033	0.968	1.028	0.970	0.955	0.959	0.950	0.972	0.963	0.996	1.010
EN	1.096	1.053	0.985	1.091	1.184	1.384	1.003	1.021	–	1.045	1.074	1.051	0.986	1.049	0.987	0.973	0.977	0.968	0.991	0.981	1.015	1.030
A-LA	1.051	1.009	0.945	1.045	1.134	1.323	0.962	0.979	0.959	–	1.029	1.007	0.946	1.006	0.947	0.933	0.937	0.929	0.951	0.941	0.973	0.987
P-LA	1.028	0.983	0.923	1.017	1.109	1.275	0.938	0.954	0.936	0.977	–	0.985	0.925	0.980	0.927	0.913	0.916	0.908	0.926	0.919	0.950	0.963
BG	1.050	1.011	0.945	1.046	1.131	1.334	0.963	0.983	0.961	1.002	1.035	–	0.944	1.008	0.946	0.933	0.937	0.929	0.954	0.942	0.972	0.988
RF	1.112	1.071	1.000	1.111	1.198	1.426	1.020	1.039	1.017	1.062	1.095	1.065	–	1.067	1.001	0.987	0.991	0.982	1.009	0.997	1.031	1.047
GB	1.049	1.007	0.942	1.042	1.132	1.310	0.959	0.976	0.957	1.000	1.026	1.006	0.944	–	0.946	0.932	0.936	0.927	0.948	0.939	0.970	0.984
NN$_1^1$	1.113	1.074	1.001	1.114	1.204	1.436	1.022	1.042	1.019	1.064	1.099	1.068	1.002	1.070	–	0.987	0.992	0.983	1.012	0.998	1.032	1.050
NN$_1^{10}$	1.128	1.088	1.015	1.129	1.218	1.457	1.035	1.055	1.032	1.078	1.112	1.083	1.015	1.084	1.014	–	1.005	0.996	1.024	1.011	1.046	1.063
NN$_2^1$	1.123	1.083	1.010	1.123	1.211	1.446	1.030	1.049	1.027	1.073	1.106	1.078	1.010	1.078	1.011	0.996	–	0.991	1.018	1.006	1.041	1.057
NN$_2^{10}$	1.133	1.093	1.020	1.134	1.222	1.459	1.039	1.058	1.036	1.082	1.115	1.088	1.020	1.088	1.020	1.005	1.009	–	1.028	1.015	1.051	1.067
NN$_3^1$	1.109	1.064	0.996	1.103	1.197	1.396	1.013	1.031	1.011	1.056	1.083	1.064	0.998	1.060	0.999	0.984	0.987	0.978	–	0.991	1.026	1.040
NN$_3^{10}$	1.117	1.075	1.005	1.114	1.205	1.424	1.023	1.042	1.021	1.066	1.096	1.072	1.005	1.070	1.006	0.991	0.995	0.986	1.011	–	1.035	1.050
NN$_4^1$	1.081	1.039	0.972	1.076	1.167	1.365	0.990	1.008	0.987	1.031	1.061	1.035	0.973	1.034	0.974	0.960	0.964	0.955	0.979	0.968	–	1.015
NN$_4^{10}$	1.067	1.024	0.958	1.060	1.150	1.337	0.976	0.993	0.973	1.016	1.044	1.022	0.960	1.019	0.961	0.947	0.951	0.942	0.964	0.955	0.986	–

Notes: We report the out-of-sample realized variance forecast MSE of each model in the selected column relative to the benchmark in the selected row. Each number is a cross-sectional average of such pairwise relative MSEs for each stock. The formatting is as follows: *number* (***number***) [___**number**___] denotes whether the Diebold–Mariano test of equal predictive accuracy is rejected more than 50% of the time at the 10% (5%) [1%] level of significance across individual tests for each asset. The hypothesis being tested is H$_0$: MSE$_i$ = MSE$_j$ against a one-sided alternative H$_1$: MSE$_i$ > MSE$_j$, where model i is the label of the selected row, whereas model j is the label of the selected column.

former, but combining RR and LA via the EN is best overall.[18] The EN ranks on top of the linear predictions with a pronounced 8.4% reduction in MSE relative to the HAR.

An appealing observation that can be made from Table 3 is that the dimension of the feature space is rather critical to achieve good performance with regression trees. In $\mathcal{M}_{\mathrm{HAR}}$, their performance were inferior to HAR, but with the expanded database they work better, although BG and GB are still worse than a regularized HAR. The best-in-class tree-based approach is the RF. It delivers a 9.9% reduction in MSE against HAR. This is arguably caused by its ability to decorrelate trees via random selection of the number of explanatory variables, as opposed to BG and GB. Comparing RF to GB, the latter shows inferior forecast accuracy. Since GB produces a sequential model based on an ensemble of weak learners, it places more weight on outlying observations in contrast to classical tree-based approaches. Hence, it is unsurprising that RF does a better job than GB, given the low signal-to-noise ratio in the data.[19] It should be stressed that even though RF has the lowest MSE among linear regression and tree-based models, there is only a marginal drop between it and the EN, which is moreover typically not statistically significant. This highlights that automatic interactions between variables can be informative about future volatility. Meanwhile, tackling overfitting is a more potent source of forecast improvement in the "high-dimensional" $\mathcal{M}_{\mathrm{ALL}}$ framework compared with $\mathcal{M}_{\mathrm{HAR}}$.

At last, we compare the NNs to HAR. It is evident that the average MSE decreases further, such that the NNs remain in front. However, the enlargement of the feature space benefits several models, and the edge of the NNs is rough as before. What is more, when additional predictors are included, the improvement is statistically significant at the 1% level across more than half of the stocks.

To shed further light on the causes of the decline in MSE for the NNs, a logical next step is to inspect their complexity. In this aspect the geometric pyramid structure is helpful, as we can readily investigate the effects of changing network architecture. Indeed, our results show that there is no big demand for an overly complex model, because the results are rather stable across network configurations. Initially, adding hidden layers to the network is helpful. Peak performance is attained with a NN_2–NN_3 architecture, but it starts to deteriorate after that. Moreover, ensemble averaging the forecast over several trained NNs leads to further declines in MSE, except again for NN_4.

Finally, we highlight that the best HAR model is the LogHAR. At first glance, it remains competitive against the ML techniques even in the $\mathcal{M}_{\mathrm{ALL}}$ setting. Thus, while a regularized HAR outperforms a non-regularized HAR, it is in turn overtaken by LogHAR. As the latter is nonlinear, this implies that handling nonlinearities is a key source to improvement, also evidenced by the ML techniques. This confirms that regularization is less vital than capturing nonlinearities, which may partially be explained by the fact that $\mathcal{M}_{\mathrm{ALL}}$ is not too vast in our setting.[20] However, in contrast to ML the true story behind the success of the LogHAR is not to be found in a general leap in forecast accuracy, but rather in a more narrow

---

18   The average value of the weight parameter α is about 0.165 in our sample.

19   We also estimated an XGBoost model of Chen and Guestrin (2016) with a refined set of hyperparameters, early stopping, and $\mathcal{L}_1$ penalization. However, the results were not much different from GB.

20   We only detect minor improvements in forecast accuracy when regularizing the LogHAR. For example, in $\mathcal{M}_{\mathrm{ALL}}$ the relative MSE of the LogHAR regularized with EN to the unregularized version
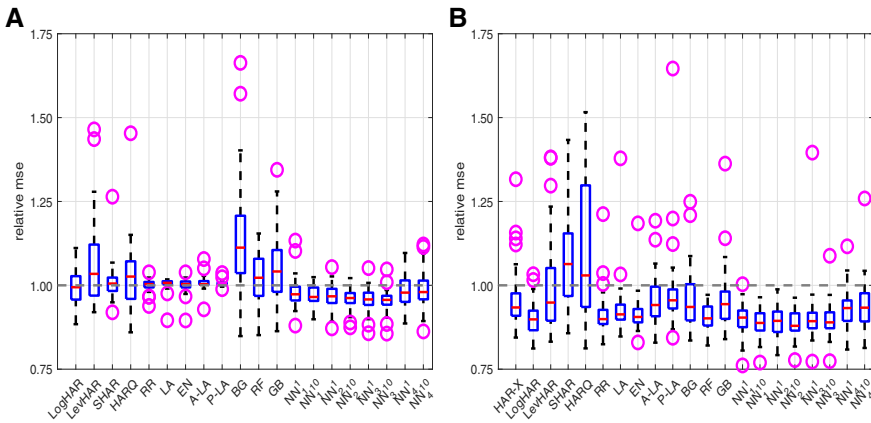
Figure 3 Boxplot of cross-sectional out-of-sample relative MSE. (**A**) $\mathcal{M}_{HAR}$. (**B**) $\mathcal{M}_{ALL}$.

Notes: We construct a boxplot for the one-day-ahead out-of-sample forecast MSE relative to the HAR model for HAR-X, LogHAR, LevHAR, SHAR, HARQ, RR, LA, EN, A-LA, P-LA, BG, RF, GB, and eight NNs. The sample consists of 29 relative MSEs for each model, corresponding to the number of stocks in our empirical application. The central mark is the median MSE, while the bottom and top edge of the box indicate the interquartile range. The whiskers are the outermost observations not flagged as outliers (the latter are marked with a circle). HAR-X is omitted from Panel A, as it is identical to HAR in $\mathcal{M}_{HAR}$.

improvement in high states of realized variance, which is a reflection of the log-transformations capability to bring down outliers. This becomes apparent when we take an in-depth look at the forecast accuracy over the entire realized variance distribution in Figure 5.

To sum up, the main improvement in forecasting accuracy in $\mathcal{M}_{ALL}$ is due to the inclusion of additional explanatory variables, which generally aids the ML techniques and the LogHAR, whereas the unregularized linear HAR models suffer from overfitting. Within the ML models, the ones that capture nonlinearities are superior, and regularization turns out to be more potent when growing the feature space.

To examine whether the superior forecast accuracy of the ML methods is driven by a small subset of stocks or it extends to the entire cross-section, we next dissect the aggregate numbers reported in Tables 2 and 3. In Figure 3, we construct a boxplot of the relative MSE for each model versus the HAR across the 29 stocks in our sample. Panel A is for $\mathcal{M}_{HAR}$, whereas Panel B is for $\mathcal{M}_{ALL}$. The tree-based approaches are slightly inferior to the HAR for $\mathcal{M}_{HAR}$, as also noticed above. However, there is a steady and sustained improvement in forecast accuracy of ML as we grow the information set.

In Figure 4, we compute the MCS at a 90% confidence level. The figure shows the percentage of times a model was retained in the final set across stocks. In line with our earlier results, the NNs exhibit an inclusion rate of around 80% for the $\mathcal{M}_{ALL}$ dataset, compared with about 20–40% for the HAR. The LogHAR is the sole HAR model that is retained at a high rate in the $\mathcal{M}_{ALL}$ setting.

is 0.995. We believe this is because the log-transformation reduces the influence of outliers and attenuates the need for penalization.
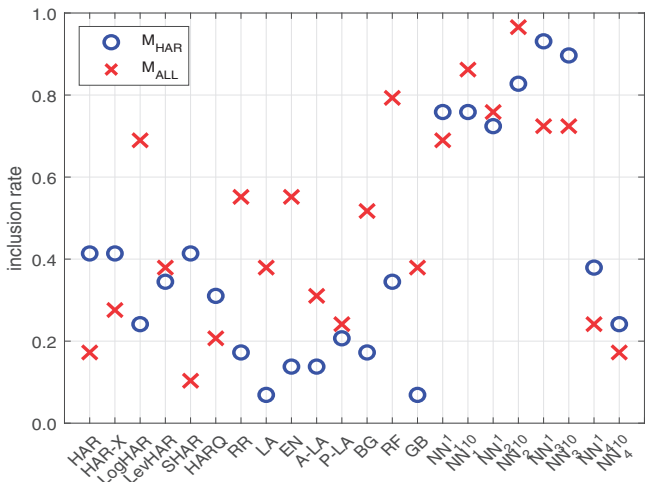
Figure 4 Inclusion rate in the MCS.

Notes: We report the outcome of the Hansen, Lunde, and Nason (2011) MCS procedure, which identifies a subset of best models by running a sequence of tests for equal predictive ability, $H_0 : \mathrm{MSE}_{\mathrm{model}_i} = \mathrm{MSE}_{\mathrm{model}_j}$. The overall confidence level is set to 90% and the numbers indicate the percentage of times a given model is retained in the MCS.

To get insights about how good the models forecast across states of the volatility distribution, we split the test set into deciles of the observed values of daily realized variance (i.e., the first subsample holds the 10% smallest values of realized variance in the out-of-sample window and is denoted (0.0,0.1)). For brevity we select a "preferred" model from each main category, that is, we look at HAR-X, LogHAR, EN, RF, and NN_2^{10}.

In Figure 5, we present the outcome of these calculations for selected low, medium, and high deciles. As indicated, the RF and NN yield lower MSE compared with the HAR across the entire support of the volatility distribution. Hence, the key driver for the ML results can be found in the majority of the sample. The MSE is only reduced by 7.38% for the highest 10% of the out-of-sample realized variance, suggesting a closer alignment in forecast accuracy during market turmoil, which is of course crucial in practice. The sustained results are not shared with the HAR-X, LogHAR, and EN. On the one hand, the HAR-X struggles when volatility is high, but has a much better MSE in tranquil markets. On the other hand, LogHAR yields only a minor improvement compared with the HAR, which appears to contradict the conclusion of Tables 2 and 3. However, a closer look reveals only a sizable improvement of the LogHAR in the highest decile. As realized variance is highly right-skewed with extreme outliers, it is therefore the ability of the LogHAR to downweigh outliers that brings its average squared error markedly down.

3.2. Marginal Effects and Variable Importance
The autonomy of a NN or tree-based approach severely limits our ability to explain what is going on "under the hood," which often leads ML to be accused of being a black box
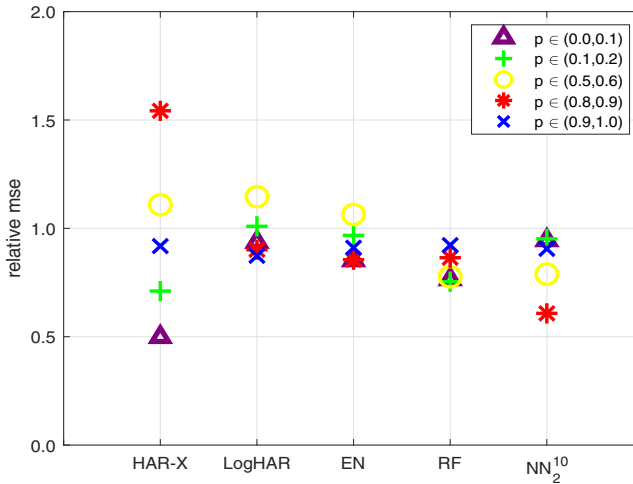
Figure 5 Forecast accuracy over out-of-sample volatility distribution.

Notes: We report one-day-ahead out-of-sample forecast MSE relative to the HAR model for HAR-X, LogHAR, EN, RF, and NN_2^{10}. The sample is split into deciles of realized variance in the test set. p stands for percentile. The dataset is $\mathcal{M}_{ALL}$.

approach. With their complex and highly nonlinear structure, a complete revelation of the underlying machinery is of course also rather difficult.

Nonetheless, we can illustrate the inner workings of the ML algorithms and shed at least some initial light on the black box by exploiting a recent contribution of Apley and Zhu (2020) that allows to compute an intuitive measure of marginal effect and variable importance in the ML framework. In particular, we calculate an ALE plot, which is an in-sample statistic based on the fitted model. To explain the concept, suppose for ease of exposition that $f(Z_t)$ is differentiable and that we are interested in analyzing the marginal effect of the jth predictor Z_{jt}, $j = 1, \ldots, J$, from the fitted model $\widehat{f}(Z_t)$. We divide Z_t into Z_{jt} and the complement $Z_{\complement t} = Z_t \setminus Z_{jt}$, that is, the remaining predictors in Z_t.

The ALE on $\widehat{f}(Z_t)$ of Z_{jt} is the function:

$$
\begin{aligned}
f^{\text{ALE}}(z) &= \text{constant} + \int_{\min(z_j)}^{z} E\left[\frac{\partial \widehat{f}(Z_{jt}, Z_{\complement t})}{\partial Z_{jt}} | Z_{jt} = x\right] dx \\
&= \text{constant} + \int_{\min(z_j)}^{z} \int \frac{\partial \widehat{f}(x, z_{\complement t})}{\partial x} p_{Z_{\complement t}|Z_{jt}}(z_{\complement t}|x) dz_{\complement t} dx,
\end{aligned}
\tag{27}
$$

where $p_{Z_{\complement t}|Z_{jt}}(z_{\complement t}|x)$ is the conditional joint density of $Z_{\complement t}$ given $Z_{jt} = x$ and the function is defined over the range of observed values of Z_{jt} in the training set, as explained below. The constant is such that $E[f^{\text{ALE}}(Z_{jt})] = 0$.

The term $\frac{\partial f(x, z_{\complement t})}{\partial x}$ is called the local effect of Z_{jt} on $\widehat{f}(Z_t)$. By accumulating the local effect, the ALE can be interpreted as the combined impact of Z_{jt} at the value z compared with the average prediction of the data. The derivative isolates the effect of the target predictor and controls for correlated features. Hence, ALE calculates changes in the prediction and adds them up, rather than computing the prediction directly. As explained in Apley and

Zhu (2020), this circumvents problems with omitted variable biases. Moreover, by exploiting the conditional distribution the ALE is not impaired by unlikely data points, a key weakness of the more common Partial Dependence (PD) function of Friedman (2001).

f^{ALE} can be estimated via a finite difference approach. The information set used in the estimation is the matrix $Z = (Z_1, \ldots, Z_{T_0})$, where T_0 is the number of days in the training sample. The observations of the jth predictor correspond to the entries in the jth row of Z, which we denote $Z_j = (z_{j1}, \ldots, z_{jT_0})$. $Z_{\complement}$ is the complement of Z_j, that is, the remaining rows in Z. We partition the range of Z_j such that $\min(Z_j) - \epsilon = z_0 < z_1 < \ldots < z_K = \max(Z_j)$, where ϵ is a small positive number ensuring that $\min(Z_j)$ is included in the subsequent calculations.[21] Next, as a function of $z \in (z_0, z_K)$ denote by $\text{idx}(z)$ the index of the interval into which z falls, that is, $z \in (z_{\text{idx}(z)-1}, z_{\text{idx}(z)}]$. The uncentered ALE is estimated as follows:

$$\tilde{f}^{\text{ALE}}(z) = \sum_{k:z_k \leq z} \frac{1}{T_k} \sum_{t:z_{jt} \in (z_{k-1}, z_k]} [\widehat{f}(z_k, z_{\complement t}) - \widehat{f}(z_{k-1}, z_{\complement t})], \tag{28}$$

where T_k is the number of data points of Z_j that fall in $(z_{k-1}, z_k]$ with $T_0 = \sum_{k=1}^{K} T_k$. As readily seen, the second summation approximates the local effect across small intervals by averaging over complement predictors $Z_{\complement}$ that are associated with the observations of $z_{jt} \in (z_{k-1}, z_k]$. The first sum accumulates these effects.

The centered ALE is then

$$\widehat{f}^{\text{ALE}}(z) = \tilde{f}^{\text{ALE}}(z) - \frac{1}{T_0} \sum_{t=1}^{T_0} \tilde{f}^{\text{ALE}}(z_{jt}). \tag{29}$$

An ALE plot is based on the graph of $(z, \widehat{f}^{\text{ALE}}(z))$.

We convert ALE to a variable importance measure following Greenwell et al. (2018) in the PD framework by computing the sample standard deviation of $\widehat{f}^{\text{ALE}}$, that is,

$$I(Z_j) = \sqrt{\frac{1}{T_0 - 1} \sum_{t=1}^{T_0} \left[\widehat{f}^{\text{ALE}}(z_{jt}) \right]^2}. \tag{30}$$

Intuitively, if our prediction is only weakly related to Z_j, once we control for the effects of other predictors, Z_j has a low importance. Consistent with this, the importance score is then $I(Z_j) \simeq 0$. On the other hand, if $\widehat{f}^{\text{ALE}}$ fluctuates a lot over its domain, it means that Z_j has a large influence on the response variable, leading to larger values of $I(Z_j)$.

Proceeding in this way, we construct a sequence $I(Z_j)$, for $j = 1, \ldots, J$. Our measure of variable importance is based on expressing each $I(Z_j)$ relative to their sum:

$$\text{VI}(Z_j) = \frac{I(Z_j)}{\sum_{j=1}^{J} I(Z_j)}, \quad \text{for } j = 1, \ldots, J. \tag{31}$$

Note that $\text{VI}(Z_j) \geq 0$ and $\sum_{j=1}^{J} \text{VI}(Z_j) = 1$.

In Figure 6, we plot the ALE on expected realized variance over the interval $[-1, 1]$ standard deviation. We restrict attention to a set of selected characteristics that are the

21   In our implementation, we partition Z_j into 100 subintervals containing equally many observations.
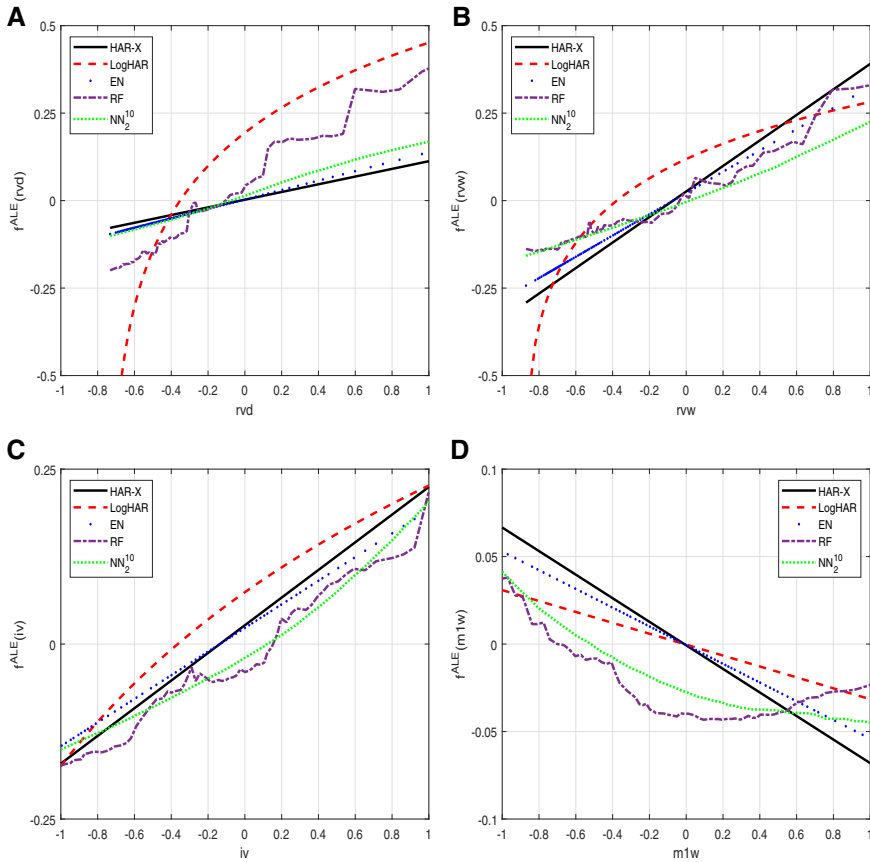
Figure 6 ALE between explanatory variable and future volatility. (**A**) RVD. (**B**) RVW. (**C**) IV. (**D**) M1W.

Notes: This figure plots the ALE between RV_t and a single explanatory variable, that is, RVD, RVW, IV, and M1W for Apple's stock price volatility. Note that the covariates are standardized. If the function does not cover the entire domain of the *x*-axis, it is because there are no observations in this area. The *y*-axis is kept constant to illuminate the relative importance of each variable in the prediction.

most influential drivers of volatility, namely RVD, RVW, IV, and M1W. Moreover, we do not explore the marginal effect averaged over stocks, as it smoothes out the unique relationship at the individual asset level. Instead, we arbitrarily select Apple for the illustration. Hence, this part serves as a showcase and may not be representative for the whole sample, although what we observe here is consistent with the broad pattern across stocks. In the construction of Figure 6, we again select a single model from each main category.

First, the figure reveals a positive relationship between RV_t and RVD, RVW, and IV. In contrast, the marginal association between RV_t and M1W is negative. This is corroborated by Table A.5 in Appendix A.3, where parameter estimates for the HAR-X model of Apple are reported. Consistent with Figure 6, the slope coefficient for M1W is negative and significant. Second, the figures illustrate a purely data-driven approach leads to a slightly

nonlinear model. Interestingly, the marginal effect for RVD detected by the RF and NN_2^{10}, especially the former, are closer to the LogHAR than the HAR. The ML algorithms, therefore, detect a stronger relationship between RV_t and RVD compared with the linear models, whereas the impact of RVW is attenuated. The discrepancy between EN and HAR-X is generally caused by the shrinkage of the parameter estimates, such that EN ends up suggesting a marginal effect that is more consistent with the RF.

The cross-sectional average VI measure is reported in Figure 7. As illustrated, there is agreement about the dominant set of predictors: RVD, RVW, and IV. Intuitively, these covariates relate to the recent history or future expectation of the stock's volatility. In contrast, less weight is assigned to variables that are not directly related to volatility. However, even if there is consensus about the best set of predictors, there are individual differences in their relative importance. It is interesting to note that IV ranks higher in the RF and NN compared with the HAR-X and EN. Hence, the ML models weigh market expectations of future volatility higher than past realizations of realized variance in forming their forecast, adding an ML perspective to the literature on the informational content of realized variance versus IV (e.g., Christensen and Prabhala 1998; Busch, Christensen, and Nielsen 2011).

With a nonlinear market structure, valuable information can potentially be extracted from a broader set of covariates. In line with this conjecture, the macroeconomic variable ADS contributes more to the prediction of volatility in the RF relative to RVM, whereas the ADS variable does not weigh much in the HAR-X. This is again consistent with the parameter estimates of the HAR-X model for Apple in Table A.5, where the coefficient in front of ADS is found to be insignificant in the $\mathcal{M}_{ALL}$ dataset.

LA applies subset selection. Accordingly, it recovers a sparse solution with a median of four beta coefficients set to zero across stocks. Thus, LA is capable of taming overfitting, but this prevents it from extracting incremental information from several of the explanatory variables, as opposed to EN, RF, or NN.

The EA variable has a low VI score. This highlights a weakness with ALE, since it is mainly a tool for finding the drivers behind the average prediction. Earnings information is released infrequently, but it is arguably very important for making short-run predictions of realized variance in the days surrounding the announcement, see Siggaard (2022).

4 Longer-Run Forecasting

In this section, we modify the forecast horizon from one-day-ahead to one-week-ahead and one-month-ahead. We replace the dependent variable in all models from next-day realized variance (RV_{t+1}) to next-week average realized variance ($RV_{t+5|t+1}$) and next-month average realized variance ($RV_{t+22|t+1}$), as a function of the information set that is available at time t, that is, Z_t. Tables 4 and 5 report the results for the weekly forecasts, whereas the monthly forecast results are available in Tables 6 and 7.

The results are striking. On the one hand, changing the forecast horizon does not alter the conclusions for $\mathcal{M}_{HAR}$ in any discernible way. The NNs are still dominating and perhaps marginally better than before. On the other hand, the picture changes for $\mathcal{M}_{ALL}$, where regression trees and NNs now provide outstanding improvements in forecast accuracy. Moreover, they appear to get better as we proceed from the weekly to monthly horizon. Whereas NNs were ahead at the one-day-ahead forecast horizon, we observe that regression trees start to be competitive at the one-week-ahead, and even surpass NNs at the

A

B

C

D
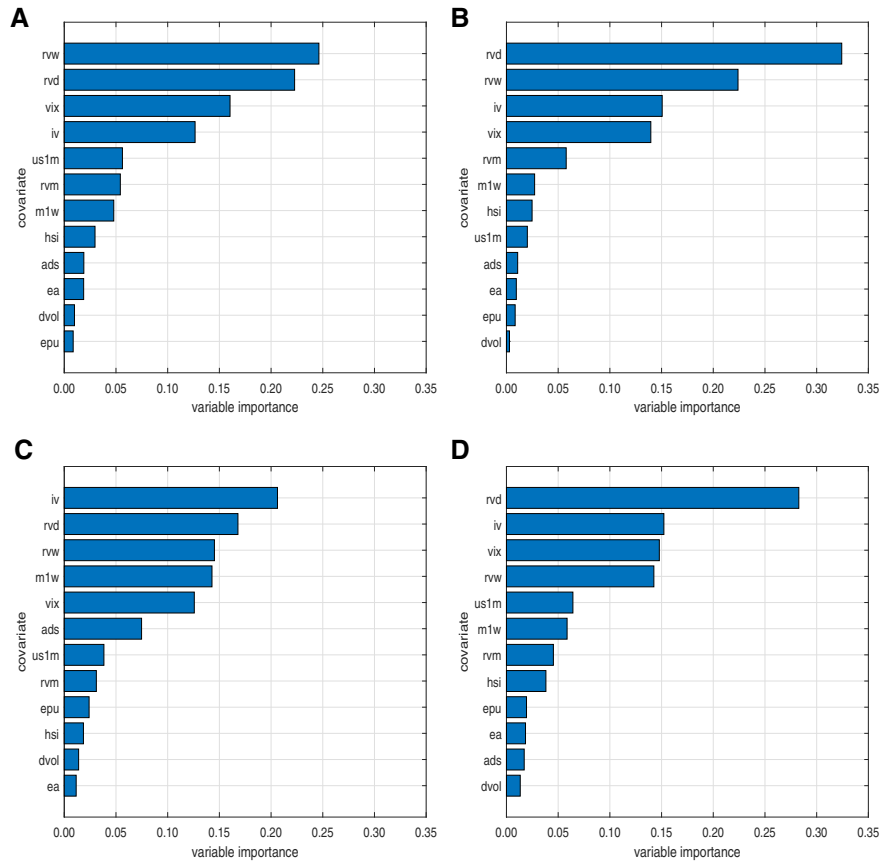
Figure 7 VI measure. (**A**) HAR-X. (**B**) EN. (**C**) RF. (**D**) NN$_2^{10}$.

Notes: We report the VI measure for each feature in the $\mathcal{M}_{\text{ALL}}$ dataset for the HAR-X, EN, RF, and NN$_2^{10}$ model. The VI measures are sorted in descending order. The figures are based on a cross-sectional average VI measure for each asset.

one-month-ahead, horizon. This shows that while nonlinearities remain important at longer horizons, functional form and interaction effects inherent in the regression trees are more crucial when expanding the predictor base and forecast horizon. The size of the reduction in relative MSE is large with a 40% or more improvement in forecast accuracy on several occasions, although one should observe that this is partly driven by a deterioration in performance for some HAR models. Even GB is outperforming most of the HAR models. The differences are often significant at the 1% level for more than 50% of the included stocks.

The LogHAR remains ahead of the HAR pack, but the Diebold–Mariano test of equal forecast accuracy with this model as the benchmark is now routinely rejected in favor of ML models. As indicated in Figure 5, the main driver behind the LogHAR model is its ability to handle outliers. As longer-run average realized variance is smoother, it is not

Table 4 One-week-ahead relative MSE and Diebold–Mariano test for dataset $\mathcal{M}_{\text{HAR}}$

	HAR	HAR-X	LogHAR	LevHAR	SHAR	HARQ	RR	LA	EN	A-LA	P-LA	BG	RF	GB	NN_1^1	NN_1^{10}	NN_2^1	NN_2^{10}	NN_3^1	NN_3^{10}	NN_4^1	NN_4^{10}
HAR	–	1.000	1.091	1.089	0.937	1.043	1.019	1.024	1.019	1.033	1.016	1.056	0.959	1.164	0.955	0.951	0.947	0.937	0.929	*0.925*	1.000	1.011
HAR-X	1.000	–	1.091	1.089	0.937	1.043	1.019	1.024	1.019	1.033	1.016	1.056	0.959	1.164	0.955	0.951	0.947	0.937	0.929	*0.925*	1.000	1.011
LogHAR	**_0.938_**	**_0.938_**	–	1.007	**0.876**	0.960	**0.948**	**0.953**	**0.949**	**0.965**	**_0.952_**	0.979	**0.888**	1.068	**_0.885_**	**0.884**	**_0.879_**	**_0.870_**	**0.862**	**_0.859_**	*0.924*	*0.934*
LevHAR	0.942	0.942	1.016	–	0.886	0.973	0.958	0.963	0.958	0.971	0.956	0.990	*0.901*	1.084	*0.896*	**0.891**	0.890	**0.878**	**0.872**	**0.867**	*0.935*	*0.944*
SHAR	1.074	1.074	1.169	1.175	–	1.118	1.092	1.099	1.093	1.109	1.092	1.128	1.024	1.253	1.023	1.020	1.013	1.004	0.995	0.991	1.074	1.087
HARQ	0.980	0.980	1.049	1.054	0.916	–	0.989	0.994	0.990	1.006	0.994	1.021	0.928	1.112	0.925	0.923	0.918	0.909	0.899	0.897	0.965	0.975
RR	**_0.990_**	**_0.990_**	1.071	1.074	*0.926*	1.023	–	1.006	1.001	1.014	1.006	1.038	0.943	1.139	**0.939**	**0.938**	0.932	0.922	**0.914**	**0.911**	0.983	0.993
LA	**_0.986_**	**_0.986_**	1.066	1.069	*0.921*	1.018	0.995	–	0.996	1.008	*1.001*	1.032	0.938	1.132	**0.934**	**0.933**	0.927	**0.917**	**0.909**	**0.906**	0.978	0.988
EN	**_0.990_**	**_0.990_**	1.071	1.074	*0.925*	1.022	0.999	1.005	–	1.014	*1.005*	1.037	0.942	1.138	**0.938**	**0.937**	0.931	**0.921**	**0.913**	**0.910**	0.982	0.992
A-LA	**_0.979_**	**_0.979_**	1.062	1.062	*0.916*	1.014	0.989	0.993	0.989	–	**0.994**	1.026	0.933	1.126	**0.929**	**0.927**	0.922	**0.911**	**0.904**	**0.901**	0.973	0.983
P-LA	**_0.984_**	**_0.984_**	1.072	1.069	0.923	1.024	1.002	1.008	1.003	1.016	–	1.037	0.943	1.142	*0.939*	*0.935*	0.932	0.921	0.913	0.909	0.983	0.993
BG	0.977	0.977	1.052	1.057	0.909	1.005	0.988	0.993	0.988	1.002	0.991	–	*0.919*	1.110	0.922	0.922	0.912	*0.905*	*0.897*	**0.893**	0.964	0.976
RF	1.060	1.060	1.142	1.149	0.986	1.092	1.072	1.078	1.073	1.089	1.076	1.097	–	1.214	1.002	1.001	0.993	0.984	0.975	0.971	1.048	1.061
GB	0.906	0.906	0.967	0.971	*0.848*	0.923	0.911	0.915	0.912	0.924	0.918	0.935	*0.855*	–	*0.853*	**0.853**	0.846	**0.837**	**0.828**	**0.827**	0.887	0.898
NN_1^1	1.058	1.058	1.140	1.144	0.988	1.090	1.071	1.076	1.071	1.087	1.074	1.104	1.005	1.216	–	0.999	0.994	0.984	0.975	0.971	1.048	1.059
NN_1^{10}	1.058	1.058	1.144	1.143	0.990	1.094	1.074	1.080	1.075	1.090	1.075	1.110	1.009	1.221	1.004	–	0.996	0.986	0.978	0.973	1.051	1.063
NN_2^1	1.067	1.067	1.152	1.156	0.995	1.101	1.080	1.086	1.081	1.096	1.083	1.110	1.012	1.225	1.010	1.008	–	0.991	0.983	**0.978**	1.056	1.068
NN_2^{10}	1.076	1.076	1.162	1.163	1.005	1.111	1.090	1.095	1.091	1.105	1.092	1.124	1.023	1.236	1.020	1.017	1.011	–	0.992	*0.988*	1.066	1.078
NN_3^1	1.087	1.087	1.173	1.177	1.015	1.120	1.100	1.106	1.101	1.117	1.103	1.134	1.032	1.245	1.030	1.027	1.021	1.011	–	0.997	1.076	1.087
NN_3^{10}	1.089	1.089	1.177	1.179	1.018	1.125	1.104	1.110	1.105	1.121	1.106	1.138	1.035	1.252	1.033	1.030	1.024	1.013	1.004	–	1.079	1.091
NN_4^1	1.017	1.017	1.093	1.095	0.952	1.044	1.028	1.034	1.029	1.045	1.032	1.061	0.965	1.158	**0.962**	*0.960*	0.954	_0.944_	0.935	_0.931_	–	1.011
NN_4^{10}	1.008	1.008	1.083	1.084	0.944	1.035	1.019	1.025	1.020	1.036	1.022	1.053	*0.957*	1.150	**0.954**	*0.952*	0.946	_0.936_	_0.927_	_0.923_	0.991	–

Notes: We report the out-of-sample realized variance forecast MSE of each model in the selected column relative to the benchmark in the selected row. Each number is a cross-sectional average of such pairwise relative MSEs for each stock. The formatting is as follows: *number* (***number***) [**_number_**] denotes whether the Diebold–Mariano test of equal predictive accuracy is rejected more than 50% of the time at the 10% (5%) [1%] level of significance across individual tests for each asset. The hypothesis being tested is $H_0 : \text{MSE}_i = \text{MSE}_j$ against a one-sided alternative $H_1 : \text{MSE}_i > \text{MSE}_j$, where model i is the label of the selected row, whereas model j is the label of the selected column.

Table 5 One-week-ahead relative MSE and Diebold–Mariano test for dataset $\mathcal{M}_{\text{ALL}}$

	HAR	HAR-X	LogHAR	LevHAR	SHAR	HARQ	RR	LA	EN	A-LA	P-LA	BG	RF	GB	NN_1^1	NN_1^{10}	NN_2^1	NN_2^{10}	NN_3^1	NN_3^{10}	NN_4^1	NN_4^{10}
HAR	–	1.037	0.970	1.076	0.913	1.459	0.954	0.958	0.925	1.026	1.033	0.840	0.796	0.930	0.877	0.857	0.865	0.846	0.882	0.855	0.913	0.945
HAR-X	1.004	–	0.951	1.027	0.917	1.321	0.924	0.929	0.899	0.986	1.006	0.835	0.791	0.919	0.870	0.853	0.859	0.842	0.867	0.847	0.898	0.923
LogHAR	1.064	1.073	–	1.104	0.966	1.439	0.986	0.992	0.958	1.059	1.076	0.884	0.838	0.975	0.924	0.904	0.908	0.891	0.921	0.898	0.952	0.979
LevHAR	0.988	0.977	0.930	–	0.902	1.272	0.903	0.909	0.879	0.963	0.985	0.819	0.777	0.902	0.854	0.838	0.843	0.827	0.850	0.831	0.881	0.903
SHAR	1.102	1.144	1.065	1.189	–	1.622	1.051	1.055	1.018	1.131	1.139	0.923	0.874	1.023	0.966	0.942	0.950	0.929	0.970	0.940	1.004	1.040
HARQ	0.856	0.825	0.790	0.839	0.781	–	0.765	0.769	0.746	0.811	0.839	0.708	0.670	0.771	0.736	0.724	0.727	0.715	0.729	0.715	0.749	0.763
RR	1.087	1.088	1.027	1.117	0.990	1.441	–	1.008	0.973	1.073	1.094	0.904	0.856	0.997	0.943	0.924	0.930	0.911	0.938	0.917	0.972	0.998
LA	1.082	1.081	1.023	1.113	0.985	1.438	0.997	–	0.968	1.065	1.087	0.900	0.852	0.989	0.938	0.919	0.924	0.906	0.934	0.912	0.966	0.993
EN	1.116	1.118	1.056	1.150	1.016	1.489	1.029	1.034	–	1.102	1.123	0.928	0.879	1.022	0.968	0.948	0.953	0.935	0.963	0.941	0.998	1.025
A-LA	1.029	1.020	0.971	1.048	0.937	1.344	0.943	0.946	0.916	–	1.028	0.854	0.809	0.937	0.891	0.873	0.878	0.861	0.887	0.866	0.917	0.942
P-LA	0.995	0.998	0.946	1.028	0.908	1.341	0.923	0.926	0.896	0.985	–	0.830	0.786	0.915	0.864	0.847	0.852	0.835	0.862	0.841	0.893	0.918
BG	1.208	1.235	1.157	1.278	1.098	1.715	1.138	1.143	1.104	1.222	1.237	–	0.952	1.109	1.052	1.028	1.035	1.013	1.052	1.023	1.092	1.130
RF	1.266	1.297	1.215	1.343	1.151	1.802	1.193	1.199	1.158	1.283	1.297	1.054	–	1.168	1.105	1.079	1.087	1.064	1.105	1.075	1.145	1.184
GB	1.099	1.117	1.048	1.156	0.999	1.530	1.030	1.033	0.999	1.102	1.121	0.911	0.866	–	0.956	0.935	0.941	0.922	0.956	0.930	0.987	1.020
NN_1^1	1.148	1.173	1.101	1.213	1.046	1.617	1.080	1.085	1.048	1.162	1.173	0.957	0.908	1.059	–	0.978	0.986	0.965	1.001	0.974	1.037	1.072
NN_1^{10}	1.175	1.206	1.130	1.249	1.070	1.679	1.109	1.114	1.076	1.194	1.204	0.980	0.930	1.087	1.024	–	1.009	0.986	1.025	0.997	1.063	1.099
NN_2^1	1.171	1.198	1.120	1.240	1.064	1.658	1.102	1.105	1.068	1.186	1.196	0.974	0.925	1.079	1.019	0.996	–	0.980	1.018	0.991	1.055	1.090
NN_2^{10}	1.194	1.227	1.147	1.271	1.086	1.712	1.127	1.132	1.093	1.215	1.223	0.996	0.944	1.104	1.041	1.016	1.023	–	1.039	1.012	1.079	1.115
NN_3^1	1.158	1.175	1.104	1.213	1.054	1.606	1.081	1.086	1.049	1.163	1.176	0.963	0.914	1.066	1.007	0.983	0.990	0.969	–	0.977	1.041	1.070
NN_3^{10}	1.182	1.206	1.131	1.247	1.076	1.663	1.109	1.114	1.076	1.193	1.206	0.984	0.934	1.089	1.029	1.005	1.012	0.990	1.026	–	1.066	1.099
NN_4^1	1.120	1.135	1.065	1.172	1.019	1.534	1.044	1.048	1.013	1.121	1.136	0.932	0.883	1.026	0.973	0.952	0.957	0.937	0.970	0.946	–	1.031
NN_4^{10}	1.094	1.100	1.035	1.133	0.996	1.467	1.012	1.017	0.983	1.087	1.104	0.910	0.862	1.001	0.949	0.928	0.933	0.914	0.942	0.921	0.974	–

Notes: We report the out-of-sample realized variance forecast MSE of each model in the selected column relative to the benchmark in the selected row. Each number is a cross-sectional average of such pairwise relative MSEs for each stock. The formatting is as follows: *number* (***number***) [**_number_**] denotes whether the Diebold–Mariano test of equal predictive accuracy is rejected more than 50% of the time at the 10% (5%) [1%] level of significance across individual tests for each asset. The hypothesis being tested is $H_0 : \text{MSE}_i = \text{MSE}_j$ against a one-sided alternative $H_1 : \text{MSE}_i > \text{MSE}_j$, where model i is the label of the selected row, whereas model j is the label of the selected column.

Table 6 One-month-ahead relative MSE and Diebold–Mariano test for dataset $\mathcal{M}_{HAR}$

	HAR	HAR-X	LogHAR	LevHAR	SHAR	HARQ	RR	LA	EN	A-LA	P-LA	BG	RF	GB	NN_1^1	NN_1^{10}	NN_2^1	NN_2^{10}	NN_3^1	NN_3^{10}	NN_4^1	NN_4^{10}
HAR	–	1.000	1.052	1.030	1.034	0.994	1.100	1.107	1.111	1.118	1.057	1.171	1.019	1.219	0.993	0.970	0.989	0.958	0.904	0.905	1.027	1.068
HAR-X	1.000	–	1.052	1.030	1.034	0.994	1.100	1.107	1.111	1.118	1.057	1.171	1.019	1.219	0.993	0.970	0.989	0.958	0.904	0.905	1.027	1.068
LogHAR	0.988	0.988	–	1.009	1.029	0.957	1.069	1.076	1.085	1.087	1.031	1.131	0.989	1.170	0.966	0.942	0.959	0.930	0.876	0.877	0.990	1.026
LevHAR	0.984	0.984	1.028	–	1.011	0.974	1.077	1.083	1.089	1.092	1.037	1.143	0.997	1.189	0.973	0.950	0.965	0.936	0.886	0.888	1.004	1.046
SHAR	0.981	0.981	1.037	1.006	–	0.979	1.080	1.087	1.091	1.097	1.038	1.137	0.995	1.193	0.973	0.950	0.965	0.937	0.885	0.889	1.013	1.053
HARQ	1.037	1.037	1.062	1.061	1.080	–	1.121	1.127	1.137	1.139	1.079	1.172	1.028	1.230	1.015	0.991	1.000	0.973	0.916	0.918	1.034	1.074
RR	0.939	0.939	0.967	0.959	0.972	0.913	–	1.007	1.021	1.016	0.966	1.077	0.937	1.109	0.921	0.899	0.911	0.883	0.831	0.833	0.937	0.975
LA	0.934	0.934	0.960	0.953	0.966	0.907	0.995	–	1.014	1.008	0.960	1.065	0.928	1.100	0.913	0.892	0.901	0.876	0.824	0.827	0.930	0.968
EN	0.911	0.911	0.947	0.934	0.944	0.896	0.986	0.992	–	1.000	0.951	1.059	0.920	1.093	0.898	0.877	0.893	0.865	0.815	0.817	0.920	0.957
A-LA	0.932	0.932	0.957	0.948	0.963	0.904	0.990	0.995	1.010	–	0.958	1.061	0.924	1.094	0.908	0.888	0.897	0.872	0.820	0.823	0.925	0.965
P-LA	0.971	0.971	1.003	0.994	1.006	0.947	1.041	1.047	1.061	1.059	–	1.116	0.971	1.157	0.958	0.935	0.947	0.919	0.865	0.867	0.974	1.013
BG	1.001	1.001	1.025	1.026	1.022	0.963	1.086	1.088	1.103	1.098	1.046	–	0.919	1.136	0.950	0.931	0.912	0.903	0.852	0.859	0.992	1.034
RF	1.060	1.060	1.094	1.086	1.088	1.026	1.149	1.152	1.166	1.162	1.106	1.113	–	1.224	1.019	0.997	0.989	0.973	0.917	0.922	1.054	1.100
GB	0.890	0.890	0.906	0.906	0.917	0.857	0.948	0.952	0.967	0.959	0.918	0.957	0.853	–	0.856	0.837	0.834	0.817	0.773	0.776	0.879	0.915
NN_1^1	1.041	1.041	1.079	1.068	1.073	1.022	1.139	1.144	1.152	1.153	1.100	1.173	1.031	1.237	–	0.986	0.996	0.971	0.915	0.918	1.053	1.097
NN_1^{10}	1.055	1.055	1.092	1.081	1.088	1.035	1.152	1.158	1.166	1.167	1.113	1.190	1.045	1.252	1.022	–	1.010	0.984	0.928	0.931	1.066	1.110
NN_2^1	1.079	1.079	1.111	1.103	1.107	1.046	1.173	1.176	1.189	1.185	1.133	1.161	1.035	1.252	1.033	1.011	–	0.985	0.930	0.936	1.078	1.125
NN_2^{10}	1.084	1.084	1.120	1.109	1.115	1.056	1.178	1.183	1.194	1.193	1.139	1.198	1.059	1.273	1.047	1.024	1.024	–	0.944	0.948	1.086	1.130
NN_3^1	1.157	1.157	1.195	1.188	1.192	1.126	1.255	1.261	1.273	1.271	1.213	1.281	1.129	1.365	1.116	1.092	1.094	1.068	–	1.007	1.154	1.200
NN_3^{10}	1.148	1.148	1.184	1.179	1.188	1.117	1.247	1.253	1.264	1.263	1.205	1.280	1.125	1.356	1.108	1.085	1.090	1.062	0.997	–	1.143	1.189
NN_4^1	1.025	1.025	1.048	1.045	1.069	0.987	1.100	1.104	1.117	1.113	1.061	1.156	1.009	1.205	0.996	0.974	0.981	0.954	0.896	0.897	–	1.043
NN_4^{10}	0.984	0.984	1.003	1.005	1.026	0.947	1.057	1.063	1.073	1.072	1.020	1.117	0.975	1.159	0.960	0.936	0.947	0.917	0.862	0.862	0.963	–

Notes: We report the out-of-sample realized variance forecast MSE of each model in the selected column relative to the benchmark in the selected row. Each number is a cross-sectional average of such pairwise relative MSEs for each stock. The formatting is as follows: *number* (***number***) [***number***] denotes whether the Diebold–Mariano test of equal predictive accuracy is rejected more than 50% of the time at the 10% (5%) [1%] level of significance across individual tests for each asset. The hypothesis being tested is $H_0 : MSE_i = MSE_j$ against a one-sided alternative $H_1 : MSE_i > MSE_j$, where model i is the label of the selected row, whereas model j is the label of the selected column.

Table 7 One-month-ahead relative MSE and Diebold–Mariano test for dataset $\mathcal{M}_{\text{ALL}}$

	HAR	HAR-X	LogHAR	LevHAR	SHAR	HARQ	RR	LA	EN	A-LA	P-LA	BG	RF	GB	NN_1^1	NN_1^{10}	NN_2^1	NN_2^{10}	NN_3^1	NN_3^{10}	NN_4^1	NN_4^{10}
HAR	–	1.290	0.919	1.328	1.312	1.569	1.111	1.378	1.107	1.386	1.363	0.621	0.604	0.829	0.800	0.789	0.816	0.786	0.788	0.795	0.902	0.964
HAR-X	0.911	–	0.790	1.021	1.021	1.166	0.898	1.023	0.890	1.029	1.018	0.551	0.541	0.738	0.714	0.708	0.736	0.711	0.699	0.707	0.800	0.836
LogHAR	1.169	1.402	–	1.438	1.430	1.672	1.212	1.470	1.203	1.476	1.459	0.703	0.690	0.938	0.913	0.905	0.941	0.906	0.894	0.903	1.013	1.072
LevHAR	0.899	0.981	0.777	–	1.000	1.142	0.882	1.003	0.874	1.008	0.998	0.543	0.533	0.726	0.703	0.697	0.725	0.701	0.689	0.696	0.788	0.822
SHAR	0.894	0.982	0.777	1.003	–	1.148	0.883	1.007	0.875	1.013	1.001	0.539	0.529	0.721	0.700	0.693	0.718	0.696	0.686	0.693	0.785	0.820
HARQ	0.832	0.885	0.706	0.903	0.905	–	0.793	0.893	0.788	0.897	0.889	0.501	0.491	0.666	0.653	0.650	0.682	0.658	0.635	0.644	0.720	0.747
RR	1.012	1.140	0.874	1.166	1.165	1.327	–	1.168	0.997	1.176	1.160	0.615	0.603	0.822	0.793	0.788	0.825	0.797	0.775	0.787	0.883	0.927
LA	0.937	0.996	0.793	1.017	1.020	1.146	0.894	–	0.885	1.005	0.999	0.563	0.552	0.750	0.728	0.725	0.760	0.734	0.712	0.722	0.812	0.845
EN	1.019	1.143	0.877	1.169	1.168	1.336	1.009	1.171	–	1.178	1.165	0.617	0.606	0.825	0.796	0.790	0.823	0.795	0.780	0.789	0.888	0.929
A-LA	0.940	0.997	0.793	1.018	1.021	1.146	0.897	0.999	0.885	–	0.997	0.565	0.553	0.753	0.731	0.728	0.763	0.737	0.716	0.724	0.813	0.845
P-LA	0.929	0.997	0.793	1.018	1.020	1.148	0.894	1.005	0.886	1.010	–	0.561	0.550	0.749	0.727	0.722	0.756	0.730	0.710	0.719	0.809	0.842
BG	1.693	2.149	1.512	2.211	2.179	2.618	1.855	2.296	1.842	2.311	2.276	–	0.988	1.337	1.326	1.305	1.340	1.302	1.305	1.315	1.495	1.601
RF	1.705	2.191	1.533	2.256	2.223	2.672	1.885	2.346	1.874	2.359	2.325	1.021	–	1.354	1.339	1.321	1.356	1.316	1.321	1.330	1.510	1.618
GB	1.299	1.679	1.158	1.729	1.701	2.041	1.434	1.796	1.425	1.809	1.783	0.768	0.753	–	1.009	0.999	1.020	0.998	1.003	1.007	1.139	1.223
NN_1^1	1.289	1.647	1.160	1.693	1.674	2.010	1.411	1.758	1.402	1.770	1.744	0.782	0.765	1.038	–	0.993	1.020	0.990	0.995	1.003	1.141	1.222
NN_1^{10}	1.298	1.655	1.171	1.702	1.680	2.018	1.422	1.767	1.413	1.779	1.751	0.785	0.770	1.048	1.013	–	1.028	0.998	1.002	1.011	1.151	1.232
NN_2^1	1.294	1.686	1.167	1.734	1.707	2.075	1.437	1.818	1.425	1.829	1.801	0.779	0.763	1.032	1.004	0.992	–	0.978	0.990	0.995	1.137	1.216
NN_2^{10}	1.318	1.715	1.189	1.765	1.740	2.103	1.463	1.847	1.452	1.857	1.829	0.797	0.780	1.061	1.028	1.015	1.031	–	1.010	1.017	1.162	1.242
NN_3^1	1.323	1.686	1.183	1.734	1.715	2.059	1.441	1.808	1.434	1.818	1.792	0.800	0.783	1.067	1.033	1.021	1.048	1.012	–	1.018	1.163	1.240
NN_3^{10}	1.300	1.656	1.161	1.703	1.681	2.020	1.418	1.774	1.408	1.782	1.757	0.785	0.769	1.045	1.015	1.003	1.024	0.993	0.990	–	1.141	1.212
NN_4^1	1.147	1.456	1.015	1.497	1.480	1.757	1.241	1.553	1.233	1.560	1.537	0.695	0.680	0.924	0.899	0.889	0.913	0.884	0.880	0.888	–	1.065
NN_4^{10}	1.099	1.346	0.963	1.382	1.368	1.607	1.159	1.423	1.149	1.428	1.408	0.665	0.652	0.886	0.861	0.851	0.872	0.845	0.840	0.845	0.955	–

Notes: We report the out-of-sample realized variance forecast MSE of each model in the selected column relative to the benchmark in the selected row. Each number is a cross-sectional average of such pairwise relative MSEs for each stock. The formatting is as follows: *number* (**number**) [**_number_**] denotes whether the Diebold–Mariano test of equal predictive accuracy is rejected more than 50% of the time at the 10% (5%) [1%] level of significance across individual tests for each asset. The hypothesis being tested is $H_0 : MSE_i = MSE_j$ against a one-sided alternative $H_1 : MSE_i > MSE_j$, where model i is the label of the selected row, whereas model j is the label of the selected column.

surprising that LogHAR cannot compete with more sophisticated models, such as RF and NNs. This also explains the mediocre results for regularization, since this is less critical here.

Why does RF suddenly emerge as the lead forecaster of future volatility at the one-month horizon? An understanding of this can help to shed light on where the improved forecast gains are extracted from. In Figure 8, we plot the in-sample autocorrelation function (ACF) of the fitted realized variance series extracted from the initial training set for the HAR, RF, and NN_2^{10}. The ACF embodies how each model perceives the persistence of the realized variance series. In Panel A, we plot the ACF corresponding to one-day-ahead forecasts, whereas Panel B is for the one-month ahead forecasts. We observe that at the short horizon, the models are roughly equally persistent, albeit a bit lower for RF. However, at the monthly horizon RF and NNs display a visibly higher ACF, suggesting they are better at approximating an underlying long-memory structure in realized variance, which is harder to detect at the one-day-ahead window, but becomes more pronounced at longer horizons.

5 A VaR Analysis

We now turn our attention toward a more economically motivated application by looking at volatility forecasting through the lens of a VaR framework. We follow Audrino, Huang, and Okhrin (2019) and construct one-day-ahead forecasts $\widehat{\text{VaR}}_{t+1}^{\alpha}$ of $\text{VaR}_{t+1}^{\alpha}$, where $\mathbb{P}(r_{t+1} \leq \text{VaR}_{t+1}^{\alpha}) = \alpha$, which are obtained by means of filtered historical simulation (Barone-Adesi, Bourgoin, and Giannopoulos 1998; Barone-Adesi, Giannopoulos, and Vosper 1999). The VaR forecasts are evaluated using a loss function that is commonly applied in quantile regression (e.g., Koenker and Bassett 1978), namely

$$\mathcal{L} = (\alpha - d_{t+1})(r_{t+1} - \widehat{\text{VaR}}_{t+1}^{\alpha}), \tag{32}$$

where $d_{t+1} = 1_{\{r_{t+1} < \widehat{\text{VaR}}_{t+1}^{\alpha}\}}$ is the "hit" function. This loss function is asymmetric and associates a weight $1 - \alpha$ to observations for which the daily log-return is beneath the forecasted quantile, which is arguably more critical in practice, especially for negative returns. Otherwise, a weight α is assigned. Throughout, we describe the results for $\alpha = 0.05$. We also investigated $\alpha = 0.01$ without any major shift in the outcome.

We also examine the exceedance probabilities of the hit sequence by conducting the likelihood ratio tests of Kupiec (1995) and Christoffersen (1998) to inspect the unconditional coverage and independence property of the VaR forecasts. The first inspects whether the empirical rejection rates are statistically speaking equal to α, on average, whereas the second test checks this assumption conditionally on the previous outcome.

A graphical representation of the VaR forecasts for Apple are presented in Figure 9 for a few models. As evident, they capture the temporal clustering of volatility. The forecast comparison and exceedance probability tests are reported in Tables 8 and 9. Overall, the main conclusions from the previous sections are still valid, although the VaR loss differentials are smaller and less statistically significant than before. Interestingly, the basic HAR model is remarkably solid and tough to beat in the VaR framework. The ML techniques are at best on par with the HAR in the $\mathcal{M}_{\text{HAR}}$ setting, but we do again observe some more substantial improvements in the $\mathcal{M}_{\text{ALL}}$ dataset, where many of the unregularized HAR models continue to struggle. Moreover, from the unconditional coverage test we see that ML is slightly worse
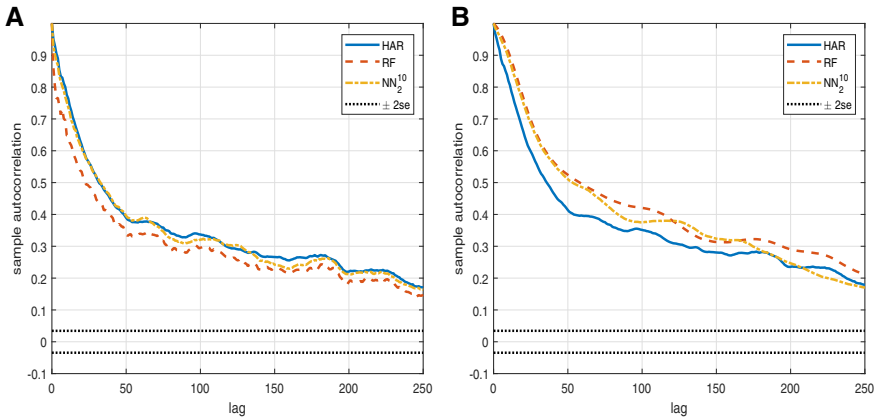
Figure 8 In-sample autocorrelation function. (**A**) One-day-ahead forecast. (**B**) One-month-ahead forecast.

Notes: We report the ACF of the in-sample fitted realized variance series at the one-day-ahead (left panel) and one-month-ahead (right panel) horizon for the HAR, RF, and NN in $\mathcal{M}_{ALL}$. The dotted lines are white noise standard error bands.
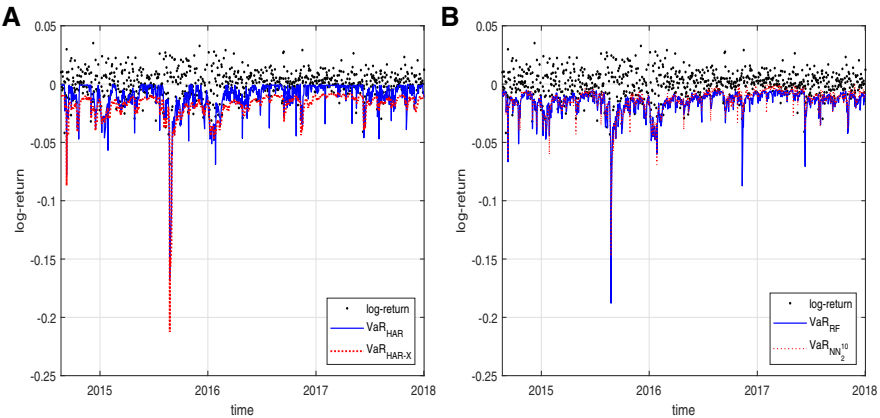
Figure 9 VaR for Apple. (**A**) HAR and HAR-X. (**B**) RF and NN_2^{10}.

Notes: We plot the open-to-close log-returns for Apple in the out-of-sample window. We superimpose the forecasted VaR at the 95% confidence level for HAR and HAR-X in Panel A, and for RF and the NN_2^{10} NN in Panel B, for $\mathcal{M}_{ALL}$.

than the HAR and LogHAR, whereas conditional on the previous outcome they are equally good. The overall impression of the results is that ML delivers precise measures of the VaR.

6 Conclusion

This article synthesizes the field of ML with volatility forecasting. We conduct an extensive out-of-sample forecast comparison of current ML algorithms with a broad suite of HAR

Table 8 One-day-ahead relative VaR and Diebold–Mariano test for dataset $\mathcal{M}_{HAR}$

	HAR	HAR-X	LogHAR	LevHAR	SHAR	HARQ	RR	LA	EN	A-LA	P-LA	BG	RF	GB	NN_1^1	NN_1^{10}	NN_2^1	NN_2^{10}	NN_3^1	NN_3^{10}	NN_4^1	NN_4^{10}
HAR	–	1.000	1.025	1.217	1.013	1.084	1.012	1.015	1.012	1.022	1.000	1.048	1.026	1.005	1.019	1.017	1.009	1.000	0.993	0.985	0.994	0.998
HAR-X	1.000	–	1.025	1.217	1.013	1.084	1.012	1.015	1.012	1.022	1.000	1.048	1.026	1.005	1.019	1.017	1.009	1.000	0.993	0.985	0.994	0.998
LogHAR	0.977	0.977	–	1.185	0.990	1.057	0.989	0.992	0.989	0.999	0.977	1.023	1.002	0.982	0.995	0.993	0.985	0.977	0.969	0.961	0.970	0.974
LevHAR	0.841	0.841	0.859	–	0.852	0.904	0.852	0.855	0.851	0.860	0.841	0.880	0.862	0.846	0.857	0.855	0.848	0.840	0.834	0.828	0.834	0.837
SHAR	0.988	0.988	1.012	1.202	–	1.071	1.000	1.003	1.000	1.009	0.987	1.035	1.014	0.993	1.007	1.004	0.996	0.988	0.981	0.973	0.982	0.986
HARQ	0.930	0.930	0.950	1.121	0.941	–	0.941	0.944	0.941	0.951	0.929	0.972	0.952	0.935	0.945	0.944	0.936	0.928	0.921	0.914	0.922	0.926
RR	0.990	0.990	1.015	1.204	1.003	1.073	–	1.003	1.000	1.009	0.990	1.037	1.016	0.993	1.009	1.006	0.998	0.990	0.983	0.975	0.984	0.987
LA	0.987	0.987	1.012	1.202	1.000	1.071	0.997	–	0.997	1.006	0.987	1.034	1.013	0.991	1.006	1.004	0.995	0.987	0.980	0.972	0.981	0.985
EN	0.990	0.990	1.015	1.205	1.003	1.074	1.000	1.003	–	1.009	0.990	1.037	1.016	0.994	1.009	1.007	0.998	0.990	0.983	0.975	0.984	0.988
A-LA	0.982	0.982	1.006	1.196	0.995	1.065	0.991	0.994	0.991	–	0.982	1.028	1.008	0.985	1.001	0.998	0.990	0.982	0.975	0.967	0.976	0.980
P-LA	1.000	1.000	1.025	1.217	1.013	1.085	1.013	1.016	1.013	1.022	–	1.048	1.027	1.006	1.020	1.017	1.009	1.001	0.993	0.985	0.994	0.998
BG	0.957	0.957	0.979	1.163	0.969	1.035	0.968	0.971	0.968	0.977	0.956	–	0.980	0.961	0.974	0.972	0.964	0.956	0.949	0.941	0.950	0.954
RF	0.976	0.976	0.999	1.186	0.988	1.056	0.988	0.991	0.988	0.997	0.975	1.020	–	0.981	0.993	0.991	0.983	0.975	0.968	0.960	0.969	0.973
GB	0.997	0.997	1.021	1.213	1.009	1.081	1.007	1.010	1.007	1.016	0.996	1.043	1.023	–	1.015	1.013	1.005	0.997	0.989	0.981	0.990	0.994
NN_1^1	0.983	0.983	1.007	1.196	0.996	1.064	0.995	0.998	0.995	1.005	0.983	1.028	1.008	0.988	–	0.998	0.990	0.982	0.975	0.967	0.976	0.980
NN_1^{10}	0.985	0.985	1.008	1.197	0.997	1.066	0.997	1.000	0.997	1.006	0.984	1.030	1.010	0.990	1.002	–	0.992	0.984	0.977	0.969	0.978	0.982
NN_2^1	0.993	0.993	1.017	1.207	1.006	1.075	1.005	1.008	1.005	1.014	0.993	1.039	1.018	0.998	1.011	1.009	–	0.992	0.985	0.977	0.986	0.990
NN_2^{10}	1.001	1.001	1.025	1.215	1.014	1.083	1.013	1.016	1.012	1.022	1.000	1.047	1.026	1.005	1.019	1.017	1.008	–	0.993	0.984	0.994	0.998
NN_3^1	1.009	1.009	1.033	1.226	1.022	1.092	1.021	1.024	1.021	1.031	1.009	1.055	1.034	1.014	1.027	1.025	1.017	1.008	–	0.992	1.002	1.006
NN_3^{10}	1.017	1.017	1.041	1.236	1.030	1.101	1.029	1.032	1.029	1.039	1.017	1.064	1.042	1.022	1.035	1.033	1.025	1.016	1.008	–	1.010	1.014
NN_4^1	1.007	1.007	1.031	1.223	1.020	1.091	1.020	1.023	1.019	1.029	1.007	1.054	1.033	1.012	1.026	1.024	1.015	1.007	0.999	0.991	–	1.004
NN_4^{10}	1.004	1.004	1.028	1.217	1.017	1.087	1.016	1.019	1.016	1.026	1.004	1.050	1.030	1.009	1.022	1.020	1.012	1.003	0.996	0.988	0.996	–
Prb.	0.053	0.053	0.047	0.110	0.057	0.086	0.041	0.044	0.041	0.046	0.050	0.070	0.069	0.040	0.073	0.071	0.067	0.065	0.062	0.056	0.039	0.032
Unc.	5	5	3	26	10	22	12	9	12	11	6	20	20	10	20	19	18	15	11	7	11	18
Cond.	4	4	1	20	2	2	4	4	4	4	4	0	2	3	2	3	2	2	2	2	2	1

Notes: We report the VaR forecast from each model in the selected column relative to the benchmark in the selected row. Each number is a cross-sectional average of such pairwise relative VaRs for each stock. The VaR confidence level is $1 - \alpha = 0.95$. The formatting is as follows: *number* (**number**) [**number**] denotes whether the Diebold–Mariano test of equal predictive accuracy is rejected more than 50% of the time at the 10% (5%) [1%] level of significance across individual tests for each asset. The hypothesis being tested is $H_0 : \mathbb{E}(\mathcal{L}_{VaR_i}) = \mathbb{E}(\mathcal{L}_{VaR_j})$ against a one-sided alternative $H_1 : \mathbb{E}(\mathcal{L}_{VaR_i}) > \mathbb{E}(\mathcal{L}_{VaR_j})$, where model i is the label of the selected row, whereas model j is the label of the selected column. Prb. is the exceedance rate, Unc. (Cond.) is the number of times the unconditional (conditional) coverage test was rejected at the 5% level of significance.

Table 9 One-day-ahead relative VaR and Diebold–Mariano test for dataset $\mathcal{M}_{\text{ALL}}$

	HAR	HAR-X	LogHAR	LevHAR	SHAR	HARQ	RR	LA	EN	A-LA	P-LA	BG	RF	GB	NN_1^1	NN_1^{10}	NN_2^1	NN_2^{10}	NN_3^1	NN_3^{10}	NN_4^1	NN_4^{10}
HAR	–	1.320	0.995	1.397	1.523	1.294	1.067	1.077	1.043	1.119	1.348	0.999	0.988	0.973	1.010	0.999	0.980	0.965	0.981	0.965	0.976	0.980
HAR-X	0.773	–	0.767	1.057	1.164	0.981	0.818	0.827	0.802	0.858	1.020	0.771	0.763	0.752	0.778	0.769	0.756	0.745	0.755	0.744	0.751	0.755
LogHAR	1.008	1.327	–	1.402	1.533	1.298	1.071	1.082	1.048	1.125	1.355	1.005	0.994	0.979	1.015	1.004	0.985	0.971	0.985	0.970	0.980	0.985
LevHAR	0.733	0.947	0.727	–	1.105	0.929	0.775	0.784	0.760	0.814	0.966	0.731	0.723	0.713	0.738	0.730	0.717	0.706	0.715	0.706	0.712	0.715
SHAR	0.664	0.868	0.661	0.919	–	0.852	0.707	0.712	0.691	0.740	0.885	0.663	0.656	0.646	0.669	0.661	0.650	0.640	0.650	0.640	0.647	0.651
HARQ	0.791	1.025	0.784	1.083	1.193	–	0.835	0.845	0.819	0.879	1.044	0.788	0.780	0.768	0.794	0.786	0.772	0.761	0.770	0.760	0.766	0.770
RR	0.944	1.237	0.937	1.308	1.434	1.209	–	1.012	0.980	1.054	1.262	0.941	0.931	0.918	0.951	0.941	0.923	0.910	0.923	0.909	0.917	0.922
LA	0.933	1.225	0.927	1.296	1.415	1.199	0.992	–	0.970	1.040	1.250	0.931	0.921	0.907	0.940	0.929	0.912	0.899	0.912	0.899	0.908	0.912
EN	0.963	1.264	0.956	1.337	1.462	1.236	1.022	1.033	–	1.074	1.290	0.960	0.950	0.936	0.970	0.959	0.941	0.927	0.941	0.927	0.936	0.941
A-LA	0.900	1.180	0.894	1.248	1.364	1.157	0.958	0.965	0.936	–	1.205	0.897	0.888	0.875	0.907	0.896	0.880	0.867	0.880	0.867	0.876	0.880
P-LA	0.760	0.982	0.754	1.038	1.143	0.962	0.804	0.813	0.787	0.844	–	0.758	0.750	0.739	0.765	0.756	0.744	0.732	0.741	0.731	0.738	0.742
BG	1.004	1.323	0.997	1.399	1.526	1.295	1.068	1.078	1.044	1.120	1.351	–	0.989	0.975	1.011	1.000	0.980	0.966	0.981	0.966	0.976	0.981
RF	1.014	1.338	1.007	1.415	1.543	1.309	1.080	1.090	1.056	1.133	1.366	1.011	–	0.985	1.022	1.011	0.991	0.977	0.992	0.977	0.987	0.992
GB	1.029	1.358	1.023	1.436	1.566	1.329	1.097	1.107	1.072	1.150	1.386	1.027	1.016	–	1.038	1.026	1.006	0.992	1.007	0.992	1.002	1.007
NN_1^1	0.994	1.307	0.987	1.383	1.509	1.278	1.057	1.067	1.033	1.109	1.334	0.990	0.980	0.966	–	0.989	0.971	0.957	0.972	0.957	0.966	0.971
NN_1^{10}	1.005	1.323	0.998	1.400	1.525	1.295	1.070	1.079	1.045	1.121	1.350	1.002	0.991	0.976	1.012	–	0.982	0.967	0.982	0.967	0.978	0.983
NN_2^1	1.025	1.351	1.018	1.430	1.559	1.321	1.091	1.101	1.066	1.144	1.380	1.021	1.010	0.996	1.032	1.020	–	0.986	1.001	0.986	0.996	1.001
NN_2^{10}	1.039	1.369	1.032	1.449	1.579	1.340	1.106	1.116	1.081	1.160	1.398	1.036	1.024	1.009	1.047	1.035	1.015	–	1.016	1.000	1.011	1.016
NN_3^1	1.024	1.347	1.017	1.425	1.558	1.317	1.089	1.099	1.065	1.143	1.375	1.021	1.010	0.995	1.032	1.020	1.000	0.986	–	0.986	0.996	1.000
NN_3^{10}	1.039	1.368	1.032	1.447	1.580	1.338	1.106	1.116	1.081	1.160	1.396	1.036	1.025	1.009	1.047	1.035	1.015	1.000	1.015	–	1.011	1.015
NN_4^1	1.030	1.354	1.022	1.433	1.566	1.323	1.094	1.106	1.071	1.150	1.383	1.027	1.016	1.000	1.037	1.026	1.006	0.992	1.006	0.991	–	1.005
NN_4^{10}	1.026	1.349	1.018	1.427	1.562	1.317	1.090	1.102	1.066	1.146	1.377	1.022	1.012	0.996	1.033	1.022	1.002	0.988	1.002	0.987	0.996	–
Prb.	0.053	0.160	0.047	0.172	0.247	0.106	0.082	0.089	0.079	0.102	0.168	0.069	0.066	0.046	0.073	0.073	0.060	0.057	0.057	0.050	0.037	0.030
Unc.	5	29	4	29	29	27	27	25	25	27	29	20	14	3	17	17	10	8	9	10	15	18
Cond.	4	5	0	8	3	2	5	5	2	9	11	1	2	2	4	2	4	3	2	2	0	0

Notes: We report the VaR forecast from each model in the selected column relative to the benchmark in the selected row. Each number is a cross-sectional average of such pairwise relative VaRs for each stock. The VaR confidence level is $1 - \alpha = 0.95$. The formatting is as follows: *number* (**number**) [__number__] denotes whether the Diebold–Mariano test of equal predictive accuracy is rejected more than 50% of the time at the 10% (5%) [1%] level of significance across individual tests for each asset. The hypothesis being tested is $H_0 : \mathbb{E}(\mathcal{L}_{\text{VaR}_i}) = \mathbb{E}(\mathcal{L}_{\text{VaR}_j})$ against a one-sided alternative $H_1 : \mathbb{E}(\mathcal{L}_{\text{VaR}_i}) > \mathbb{E}(\mathcal{L}_{\text{VaR}_j})$, where model i is the label of the selected row, whereas model j is the label of the selected column. Prb. is the exceedance rate, Unc. (Cond.) is the number of times the unconditional (conditional) coverage test was rejected at the 5% level of significance.

models for forecasting realized variance. We show that ML is better at handling the highly nonlinear structure governing financial markets, and that they are able to extract valuable incremental information about future volatility in a data-rich environment with a large feature space; a setting in which standard HAR models struggle. The RF and NNs are preferred among ML algorithms and significantly improve prediction relative to HAR. The improvements increase at longer horizons and are often statistically significant. We dissect the underlying structure of the data via ALE plots in order to pin down the main covariates leading to the improved prediction. In an economic VaR application, we continue to observe reduced loss with ML models, although it is less statistically significant.

To the best of our knowledge, this study is the first to extensively implement a broad selection of the most widely applied ML methods for the problem of volatility prediction and compare them to a broad suite of HAR models. We envision the paper can serve as a starting point for applying ML to forecast stochastic volatility and perhaps even as a benchmark when building more complicated models. Moreover, we look forward to behold how much additional improvement that can be had by doing more extensive hyperparameter tuning than what we do in this article.

Appendix A

A.1 Span of Test Set

This section reports the out-of-sample relative MSE of each volatility model when the length of the training set is shortened from 70% of the sample, or 2,964 days, to 1,000 days (Tables A.1 and A.2) and 2,000 days (Tables A.3 and A.4), while keeping the validation set fixed at 200 days, yielding a one-for-one extension of the out-of-sample test set.

Training set = 1,000 days

Table A.1 One-day-ahead relative MSE and Diebold–Mariano test for dataset $\mathcal{M}_{\mathrm{HAR}}$

	HAR	HAR-X	LogHAR	LevHAR	SHAR	HARQ	RR	LA	EN	A-LA	P-LA	BG	RF	GB	NN_1^1	NN_1^{10}	NN_2^1	NN_2^{10}	NN_3^1	NN_3^{10}	NN_4^1	NN_4^{10}
HAR	–	1.000	0.905	0.940	1.011	0.870	0.865	0.866	0.864	0.869	0.865	1.050	0.958	0.937	0.866	0.835	0.867	0.844	0.865	0.861	0.900	0.899
HAR-X	1.000	–	0.905	0.940	1.011	0.870	0.865	0.866	0.864	0.869	0.865	1.050	0.958	0.937	0.866	0.835	0.867	0.844	0.865	0.861	0.900	0.899
LogHAR	1.169	1.169	–	1.096	1.187	0.993	0.984	0.986	0.983	0.989	0.985	1.201	1.094	1.060	0.979	0.947	0.978	0.954	0.975	0.972	1.014	1.012
LevHAR	1.066	1.066	0.965	–	1.076	0.926	0.920	0.921	0.919	0.924	0.920	1.118	1.020	0.996	0.921	0.888	0.924	0.899	0.920	0.917	0.958	0.956
SHAR	0.994	0.994	0.903	0.934	–	0.867	0.861	0.862	0.860	0.865	0.862	1.044	0.954	0.933	0.865	0.832	0.865	0.842	0.862	0.859	0.897	0.896
HARQ	1.180	1.180	1.043	1.105	1.196	–	1.003	1.004	1.002	1.008	1.003	1.216	1.111	1.079	0.996	0.965	1.003	0.975	0.998	0.994	1.036	1.035
RR	1.189	1.189	1.049	1.115	1.205	1.020	–	1.002	0.999	1.005	1.001	1.233	1.125	1.090	1.007	0.972	1.009	0.982	1.005	1.001	1.045	1.043
LA	1.186	1.186	1.048	1.112	1.203	1.017	0.998	–	0.998	1.003	0.999	1.231	1.123	1.088	1.005	0.970	1.007	0.980	1.003	0.999	1.043	1.041
EN	1.190	1.190	1.050	1.115	1.206	1.020	1.001	1.003	–	1.006	1.002	1.234	1.126	1.091	1.008	0.973	1.010	0.983	1.006	1.002	1.046	1.044
A-LA	1.182	1.182	1.044	1.109	1.198	1.014	0.995	0.997	0.994	–	0.996	1.228	1.119	1.085	1.002	0.967	1.004	0.977	1.000	0.996	1.040	1.038
P-LA	1.188	1.188	1.049	1.114	1.204	1.018	1.000	1.001	0.999	1.004	–	1.233	1.124	1.089	1.006	0.971	1.008	0.981	1.004	1.000	1.044	1.042
BG	0.977	0.977	0.861	0.915	0.988	0.834	0.833	0.834	0.832	0.837	0.833	–	0.915	0.894	0.829	0.801	0.830	0.808	0.827	0.824	0.860	0.859
RF	1.065	1.065	0.940	0.999	1.078	0.911	0.908	0.909	0.907	0.913	0.909	1.095	–	0.976	0.905	0.873	0.905	0.882	0.902	0.899	0.938	0.937
GB	1.103	1.103	0.966	1.033	1.119	0.937	0.932	0.933	0.931	0.937	0.932	1.134	1.035	–	0.927	0.897	0.929	0.905	0.926	0.922	0.962	0.960
NN_1^1	1.203	1.203	1.055	1.128	1.224	1.023	1.017	1.018	1.016	1.022	1.017	1.240	1.132	1.096	–	0.977	1.013	0.986	1.010	1.006	1.049	1.047
NN_1^{10}	1.228	1.228	1.079	1.152	1.248	1.050	1.040	1.041	1.039	1.045	1.040	1.270	1.158	1.122	1.034	–	1.036	1.010	1.034	1.030	1.075	1.073
NN_2^1	1.197	1.197	1.043	1.124	1.216	1.023	1.012	1.014	1.011	1.017	1.013	1.234	1.125	1.089	1.005	0.971	–	0.980	1.002	0.999	1.042	1.040
NN_2^{10}	1.220	1.220	1.066	1.144	1.239	1.041	1.031	1.032	1.030	1.036	1.031	1.258	1.147	1.111	1.025	0.991	1.026	–	1.023	1.019	1.063	1.062
NN_3^1	1.194	1.194	1.041	1.120	1.212	1.018	1.008	1.009	1.007	1.013	1.008	1.230	1.122	1.086	1.002	0.970	1.003	0.978	–	0.996	1.039	1.038
NN_3^{10}	1.198	1.198	1.045	1.124	1.217	1.022	1.012	1.013	1.011	1.017	1.012	1.235	1.126	1.090	1.006	0.973	1.007	0.981	1.004	–	1.043	1.041
NN_4^1	1.152	1.152	1.003	1.080	1.170	0.979	0.972	0.973	0.971	0.977	0.972	1.185	1.081	1.046	0.965	0.934	0.966	0.942	0.963	0.959	–	0.999
NN_4^{10}	1.154	1.154	1.004	1.082	1.172	0.981	0.972	0.974	0.972	0.977	0.973	1.188	1.083	1.047	0.966	0.935	0.967	0.943	0.965	0.961	1.002	–

Notes: We report the out-of-sample realized variance forecast MSE of each model in the selected column relative to the benchmark in the selected row. Each number is a cross-sectional average of such pairwise relative MSEs for each stock. The formatting is as follows: *number* (**number**) [**number**] denotes whether the Diebold–Mariano test of equal predictive accuracy is rejected more than 50% of the time at the 10% (5%) [1%] level of significance across individual tests for each asset. The hypothesis being tested is $H_0 : MSE_i = MSE_j$ against a one-sided alternative $H_1 : MSE_i > MSE_j$, where model i is the label of the selected row, whereas model j is the label of the selected column.

Table A.2 One-day-ahead relative MSE and Diebold–Mariano test for dataset $\mathcal{M}_{ALL}$

	HAR	HAR-X	LogHAR	LevHAR	SHAR	HARQ	RR	LA	EN	A-LA	P-LA	BG	RF	GB	NN_1^1	NN_1^{10}	NN_2^1	NN_2^{10}	NN_3^1	NN_3^{10}	NN_4^1	NN_4^{10}
HAR	–	0.937	0.970	0.917	0.969	0.905	0.829	0.835	0.827	0.837	1.731	0.919	0.849	0.904	*0.835*	*0.802*	0.843	0.849	0.862	0.827	0.930	0.871
HAR-X	1.075	–	1.033	0.978	1.033	0.965	0.888	0.894	0.886	0.896	1.845	0.983	0.908	0.965	0.897	0.860	0.902	0.912	0.922	0.886	0.996	0.932
LogHAR	1.169	1.086	–	1.063	1.131	1.032	0.943	0.950	0.941	0.953	1.898	1.057	0.973	1.025	0.967	0.919	0.958	0.981	0.980	0.944	1.069	0.992
LevHAR	1.101	1.024	1.061	–	1.057	0.988	0.908	0.915	0.906	0.917	1.888	1.005	0.929	0.987	0.918	0.880	0.923	0.932	0.944	0.907	1.020	0.954
SHAR	1.046	0.973	1.011	0.951	–	0.941	0.866	0.872	0.863	0.874	1.805	0.957	0.885	0.941	0.876	0.839	0.882	0.889	0.899	0.864	0.973	0.909
HARQ	1.138	1.056	1.074	1.032	1.095	–	0.927	0.933	0.925	0.935	1.898	1.030	0.949	1.006	0.933	0.895	0.937	0.950	0.960	0.922	1.035	0.970
RR	1.238	1.157	1.169	1.131	1.200	1.105	–	1.007	0.997	1.011	2.063	1.125	1.037	1.097	1.020	0.976	1.021	1.028	1.048	1.003	1.134	1.055
LA	1.231	1.151	1.161	1.125	1.194	1.099	0.994	–	*0.991*	1.004	2.049	1.118	1.031	1.091	1.013	0.970	1.014	1.020	1.041	0.997	1.126	1.048
EN	1.242	1.161	1.172	1.134	1.204	1.108	1.003	1.009	–	1.013	2.068	1.127	1.040	1.100	1.023	0.979	1.024	1.031	1.051	1.006	1.137	1.058
A-LA	1.225	1.145	1.158	1.119	1.188	1.093	0.990	0.997	*0.988*	–	2.043	1.113	1.026	1.086	1.009	0.966	1.011	1.016	1.037	0.993	1.121	1.044
P-LA	_**0.657**_	_**0.605**_	_**0.582**_	_**0.592**_	_**0.634**_	_**0.566**_	_**0.518**_	_**0.521**_	_**0.516**_	_**0.523**_	–	_**0.586**_	_**0.536**_	_**0.562**_	_**0.531**_	_**0.504**_	_**0.520**_	_**0.525**_	_**0.538**_	_**0.514**_	_**0.585**_	_**0.539**_
BG	1.105	1.030	1.060	1.007	1.065	0.987	0.904	0.911	0.902	0.913	1.873	–	*0.925*	0.985	*0.910*	*0.874*	0.918	*0.931*	0.938	0.902	1.010	0.949
RF	1.196	1.114	1.140	1.088	1.154	1.065	0.977	0.983	0.974	0.986	2.013	1.083	–	1.063	0.982	0.944	0.992	1.003	1.014	0.974	1.090	1.024
GB	1.135	1.057	1.065	1.033	1.098	1.007	*0.922*	*0.928*	*0.919*	0.930	1.887	1.029	0.948	–	***0.933***	*0.894*	0.935	*0.941*	0.958	0.919	1.035	0.966
NN_1^1	1.256	1.175	1.198	1.148	1.220	1.118	1.021	1.028	1.019	1.031	2.106	1.139	1.050	1.114	–	0.977	1.039	1.034	1.055	1.012	1.112	1.063
NN_1^{10}	1.280	1.196	1.215	1.170	1.242	1.139	1.041	1.047	1.038	1.051	2.140	1.161	1.070	1.135	1.033	–	1.058	1.059	1.077	1.033	1.147	1.086
NN_2^1	1.236	1.153	1.163	1.128	1.199	1.095	1.000	1.007	0.997	1.010	2.038	1.119	1.032	1.092	1.013	0.971	–	1.021	1.039	0.997	1.124	1.047
NN_2^{10}	1.250	1.168	1.191	1.142	1.213	1.113	1.014	1.020	1.011	1.023	2.073	1.136	1.046	1.106	1.013	0.977	1.031	–	1.050	1.000	1.125	1.052
NN_3^1	1.197	1.116	1.125	1.092	1.158	1.063	0.973	0.979	0.970	0.982	1.994	1.084	1.001	1.060	0.971	0.938	0.985	**0.990**	–	*0.964*	1.071	1.013
NN_3^{10}	1.246	1.163	1.179	1.137	1.207	1.108	1.011	1.017	1.008	1.020	2.067	1.130	1.042	1.102	1.011	0.975	1.026	1.017	1.044	–	1.119	1.051
NN_4^1	1.145	1.066	1.083	1.043	1.108	1.015	*0.929*	0.935	*0.927*	0.937	1.901	1.035	0.953	1.011	*0.910*	***0.888***	***0.943***	***0.942***	0.952	***0.916***	–	0.961
NN_4^{10}	1.189	1.109	1.120	1.084	1.152	1.055	0.963	0.969	0.961	0.972	1.965	1.077	0.992	1.050	0.962	*0.928*	0.977	***0.969***	0.995	_**0.952**_	1.064	–

Notes: We report the out-of-sample realized variance forecast MSE of each model in the selected column relative to the benchmark in the selected row. Each number is a cross-sectional average of such pairwise relative MSEs for each stock. The formatting is as follows: *number* (***number***) [_**number**_] denotes whether the Diebold–Mariano test of equal predictive accuracy is rejected more than 50% of the time at the 10% (5%) [1%] level of significance across individual tests for each asset. The hypothesis being tested is $H_0 : MSE_i = MSE_j$ against a one-sided alternative $H_1 : MSE_i > MSE_j$, where model i is the label of the selected row, whereas model j is the label of the selected column.

Training set = 2,000 days

Table A.3 One-day-ahead relative MSE and Diebold–Mariano test for dataset $\mathcal{M}_{\text{HAR}}$

	HAR	HAR-X	LogHAR	LevHAR	SHAR	HARQ	RR	LA	EN	A-LA	P-LA	BG	RF	GB	NN^1_1	NN^{10}_1	NN^1_2	NN^{10}_2	NN^1_3	NN^{10}_3	NN^1_4	NN^{10}_4
HAR	–	1.000	1.025	1.143	1.020	1.013	0.984	1.001	0.986	1.010	1.008	1.210	1.034	1.065	1.006	0.983	1.064	1.008	1.059	1.050	1.120	1.137
HAR-X	1.000	–	1.025	1.143	1.020	1.013	0.984	1.001	0.986	1.010	1.008	1.210	1.034	1.065	1.006	0.983	1.064	1.008	1.059	1.050	1.120	1.137
LogHAR	0.984	0.984	–	1.110	1.005	0.992	0.967	0.984	0.969	0.994	0.992	1.188	1.016	1.046	0.989	0.967	1.042	0.990	1.036	1.029	1.092	1.110
LevHAR	0.904	0.904	0.916	–	0.919	0.906	0.888	0.903	0.891	0.912	0.911	1.079	0.930	0.960	0.907	0.887	0.950	0.908	0.948	0.941	0.994	1.011
SHAR	0.985	0.985	1.011	1.124	–	0.996	0.969	0.986	0.972	0.995	0.993	1.186	1.015	1.047	0.988	0.967	1.043	0.992	1.040	1.032	1.101	1.117
HARQ	0.999	0.999	1.019	1.129	1.016	–	0.981	0.998	0.984	1.008	1.007	1.194	1.026	1.059	1.000	0.979	1.058	1.004	1.054	1.045	1.115	1.132
RR	1.018	1.018	1.041	1.161	1.038	1.030	–	1.018	1.003	1.028	1.026	1.229	1.051	1.082	1.023	1.000	1.080	1.025	1.075	1.066	1.136	1.153
LA	1.000	1.000	1.024	1.141	1.020	1.012	0.983	–	0.986	1.010	1.008	1.207	1.032	1.063	1.005	0.983	1.062	1.007	1.057	1.048	1.117	1.134
EN	1.015	1.015	1.039	1.159	1.035	1.028	0.997	1.015	–	1.025	1.023	1.227	1.048	1.079	1.020	0.998	1.078	1.022	1.072	1.063	1.133	1.150
A-LA	0.990	0.990	1.014	1.130	1.010	1.002	0.974	0.990	0.976	–	0.998	1.195	1.023	1.053	0.995	0.973	1.052	0.998	1.048	1.039	1.108	1.124
P-LA	0.992	0.992	1.017	1.134	1.011	1.006	0.976	0.993	0.979	1.003	–	1.201	1.026	1.057	0.998	0.976	1.056	1.000	1.051	1.042	1.111	1.128
BG	0.853	0.853	0.871	0.962	0.866	0.854	0.837	0.851	0.839	0.860	0.860	–	0.870	0.900	0.855	0.836	0.900	0.856	0.900	0.890	0.951	0.965
RF	0.976	0.976	0.998	1.110	0.992	0.982	0.958	0.975	0.961	0.985	0.984	1.165	–	1.032	0.979	0.958	1.032	0.980	1.028	1.019	1.087	1.103
GB	0.949	0.949	0.971	1.082	0.966	0.957	0.932	0.948	0.935	0.958	0.957	1.138	0.975	–	0.952	0.932	1.004	0.953	1.000	0.991	1.056	1.071
NN^1_1	0.998	0.998	1.022	1.137	1.015	1.007	0.981	0.998	0.984	1.008	1.006	1.203	1.029	1.060	–	0.979	1.057	1.003	1.052	1.044	1.114	1.130
NN^{10}_1	1.019	1.019	1.043	1.161	1.037	1.029	1.002	1.019	1.004	1.029	1.027	1.228	1.051	1.083	1.022	–	1.080	1.025	1.076	1.067	1.138	1.155
NN^1_2	0.960	0.960	0.980	1.087	0.975	0.968	0.943	0.960	0.945	0.969	0.968	1.152	0.986	1.016	0.962	0.941	–	0.963	1.002	0.995	1.053	1.069
NN^{10}_2	0.995	0.995	1.018	1.134	1.013	1.005	0.978	0.995	0.980	1.005	1.003	1.199	1.025	1.055	0.998	0.977	1.052	–	1.047	1.039	1.106	1.123
NN^1_3	0.959	0.959	0.978	1.087	0.976	0.968	0.942	0.959	0.944	0.968	0.967	1.157	0.987	1.016	0.961	0.941	1.004	0.962	–	0.995	1.053	1.069
NN^{10}_3	0.964	0.964	0.984	1.092	0.981	0.972	0.946	0.963	0.949	0.973	0.971	1.160	0.991	1.020	0.966	0.945	1.010	0.966	1.007	–	1.060	1.076
NN^1_4	0.919	0.919	0.934	1.033	0.937	0.928	0.902	0.918	0.904	0.928	0.926	1.109	0.945	0.973	0.922	0.902	0.959	0.921	0.957	0.951	–	1.017
NN^{10}_4	0.904	0.904	0.920	1.019	0.921	0.912	0.887	0.903	0.890	0.913	0.912	1.091	0.930	0.956	0.907	0.887	0.943	0.906	0.941	0.935	0.985	–

Notes: We report the out-of-sample realized variance forecast MSE of each model in the selected column relative to the benchmark in the selected row. Each number is a cross-sectional average of such pairwise relative MSEs for each stock. The formatting is as follows: *number* (**number**) [**number**] denotes whether the Diebold–Mariano test of equal predictive accuracy is rejected more than 50% of the time at the 10% (5%) [1%] level of significance across individual tests for each asset. The hypothesis being tested is $H_0 : \text{MSE}_i = \text{MSE}_j$ against a one-sided alternative $H_1 : \text{MSE}_i > \text{MSE}_j$, where model i is the label of the selected row, whereas model j is the label of the selected column.

Table A.4 One-day-ahead relative MSE and Diebold–Mariano test for dataset $\mathcal{M}_{\text{ALL}}$

	HAR	HAR-X	LogHAR	LevHAR	SHAR	HARQ	RR	LA	EN	A-LA	P-LA	BG	RF	GB	NN_1^1	NN_1^{10}	NN_2^1	NN_2^{10}	NN_3^1	NN_3^{10}	NN_4^1	NN_4^{10}
HAR	–	1.048	*0.939*	1.115	1.049	1.467	*0.976*	0.984	*0.950*	1.010	1.041	0.996	*0.918*	0.997	1.048	*0.933*	1.061	0.996	1.126	1.083	1.203	1.183
HAR-X	0.972	–	*0.905*	1.056	1.001	1.323	*0.937*	*0.945*	*0.915*	0.970	0.995	0.968	*0.893*	0.961	1.013	*0.907*	1.019	0.958	1.077	1.031	1.134	1.118
LogHAR	1.075	1.116	–	1.180	1.118	1.507	1.040	1.051	1.015	1.080	1.110	1.071	0.988	1.068	1.123	1.003	1.133	1.064	1.199	1.151	1.276	1.258
LevHAR	0.934	*0.955*	*0.865*	–	0.955	1.240	*0.894*	*0.904*	*0.876*	0.928	0.951	0.927	*0.856*	0.920	0.970	*0.871*	0.977	0.918	1.031	0.987	1.087	1.071
SHAR	0.974	1.002	*0.907*	1.058	–	1.326	*0.937*	0.946	*0.916*	0.971	0.996	0.968	*0.893*	0.961	1.013	*0.908*	1.020	0.959	1.080	1.033	1.136	1.120
HARQ	*0.826*	*0.830*	*0.758*	*0.864*	*0.831*	–	*0.781*	*0.790*	*0.768*	*0.813*	*0.827*	*0.822*	*0.759*	*0.808*	*0.851*	*0.771*	*0.856*	*0.803*	0.898	0.853	0.929	0.919
RR	1.037	1.072	0.966	1.132	1.072	1.430	–	1.010	0.977	1.038	1.065	1.032	0.952	1.026	1.081	*0.966*	1.089	1.025	1.153	1.106	1.219	1.202
LA	1.027	1.062	*0.958*	1.125	1.062	1.422	0.992	–	*0.968*	1.027	1.055	1.022	0.942	1.016	1.071	*0.957*	1.079	1.015	1.144	1.095	1.204	1.188
EN	1.060	1.100	0.989	1.165	1.100	1.484	1.025	1.034	–	1.062	1.092	1.054	0.972	1.050	1.106	0.988	1.115	1.049	1.182	1.134	1.251	1.233
A-LA	1.001	1.036	*0.935*	1.097	1.036	1.398	*0.968*	*0.975*	*0.944*	–	1.029	0.996	*0.918*	0.990	1.044	*0.933*	1.053	0.990	1.118	1.070	1.175	1.160
P-LA	0.978	1.007	*0.911*	1.065	1.008	1.332	*0.942*	*0.950*	*0.920*	0.975	–	0.974	*0.897*	0.966	1.018	*0.911*	1.024	0.964	1.084	1.038	1.137	1.124
BG	1.024	1.073	0.961	1.138	1.072	1.512	0.998	1.007	0.971	1.032	1.066	–	*0.932*	1.017	1.071	0.954	1.081	1.020	1.148	1.111	1.235	1.215
RF	1.095	1.149	1.030	1.222	1.148	1.624	1.069	1.077	1.040	1.104	1.141	1.081	–	1.090	1.147	1.020	1.160	1.092	1.233	1.190	1.320	1.299
GB	1.012	1.050	*0.947*	1.113	1.049	1.426	0.980	0.988	0.955	1.014	1.043	1.004	*0.927*	–	1.056	*0.944*	1.062	1.000	1.124	1.081	1.193	1.175
NN_1^1	0.966	1.005	*0.904*	1.065	1.005	1.364	*0.937*	*0.945*	*0.913*	0.970	0.998	0.961	*0.886*	0.959	–	*0.898*	1.019	0.957	1.078	1.036	1.143	1.128
NN_1^{10}	1.077	1.128	1.011	1.200	1.128	1.576	1.050	1.058	1.022	1.086	1.120	1.072	0.987	1.073	1.125	–	1.141	1.073	1.213	1.166	1.290	1.271
NN_2^1	0.968	1.003	*0.904*	1.064	1.004	1.344	*0.936*	*0.944*	*0.913*	0.970	0.995	0.960	*0.888*	0.956	1.010	*0.902*	–	*0.952*	1.058	1.024	1.128	1.112
NN_2^{10}	1.015	1.052	*0.948*	1.116	1.053	1.415	0.983	0.991	*0.959*	1.018	1.045	1.012	*0.933*	1.005	1.059	*0.947*	1.062	–	1.121	1.078	1.191	1.173
NN_3^1	0.928	0.957	*0.863*	1.015	0.960	1.266	*0.896*	*0.904*	*0.874*	0.930	0.952	*0.923*	*0.852*	0.916	*0.966*	*0.866*	*0.957*	*0.908*	–	0.974	1.076	1.061
NN_3^{10}	0.954	0.982	*0.888*	1.041	0.984	1.285	*0.919*	*0.927*	*0.898*	0.954	0.976	*0.952*	*0.877*	0.941	0.993	*0.890*	0.990	*0.935*	1.041	–	1.101	1.086
NN_4^1	*0.884*	*0.907*	*0.824*	*0.964*	*0.909*	1.180	*0.851*	*0.856*	*0.831*	*0.879*	*0.899*	*0.881*	*0.812*	*0.871*	*0.918*	*0.823*	*0.917*	*0.867*	*0.969*	*0.927*	–	0.995
NN_4^{10}	*0.890*	*0.912*	*0.829*	*0.968*	*0.914*	1.188	*0.856*	*0.862*	*0.836*	*0.885*	*0.906*	*0.888*	*0.818*	*0.876*	*0.925*	*0.829*	*0.922*	*0.871*	*0.972*	*0.931*	1.013	–

Notes: We report the out-of-sample realized variance forecast MSE of each model in the selected column relative to the benchmark in the selected row. Each number is a cross-sectional average of such pairwise relative MSEs for each stock. The formatting is as follows: *number* (**number**) [**_number_**] denotes whether the Diebold–Mariano test of equal predictive accuracy is rejected more than 50% of the time at the 10% (5%) [1%] level of significance across individual tests for each asset. The hypothesis being tested is $H_0 : MSE_i = MSE_j$ against a one-sided alternative $H_1 : MSE_i > MSE_j$, where model i is the label of the selected row, whereas model j is the label of the selected column.

A.2 Additional VI Measures

In the figures below, we report the VI measures for those forecasting models that were excluded from Figure 7 in Section 3.2 in the main text.
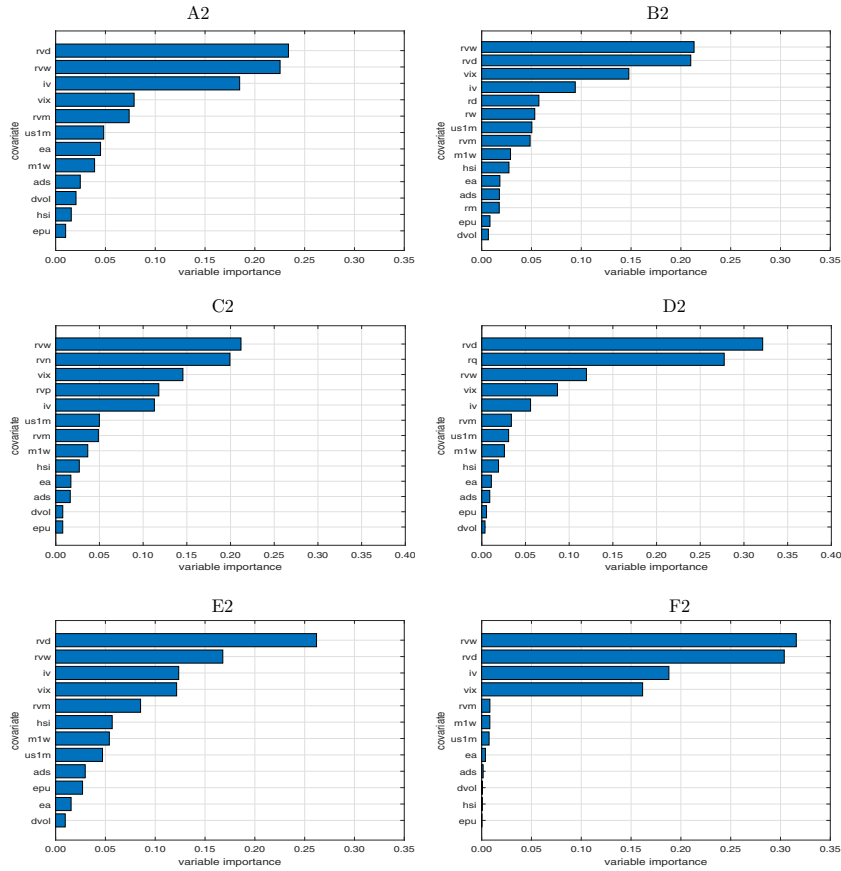

Figure A.1 VI measure. (**A**) LogHAR. (**B**) LevHAR. (**C**) SHAR. (**D**) HARQ. (**E**) RR. (**F**) LA.

Notes: We report the VI measure for each feature of LogHAR, LevHAR, SHAR, HARQ, RR, and LA, sorted according to the numeric value of VI. Refer to the main text for further details.
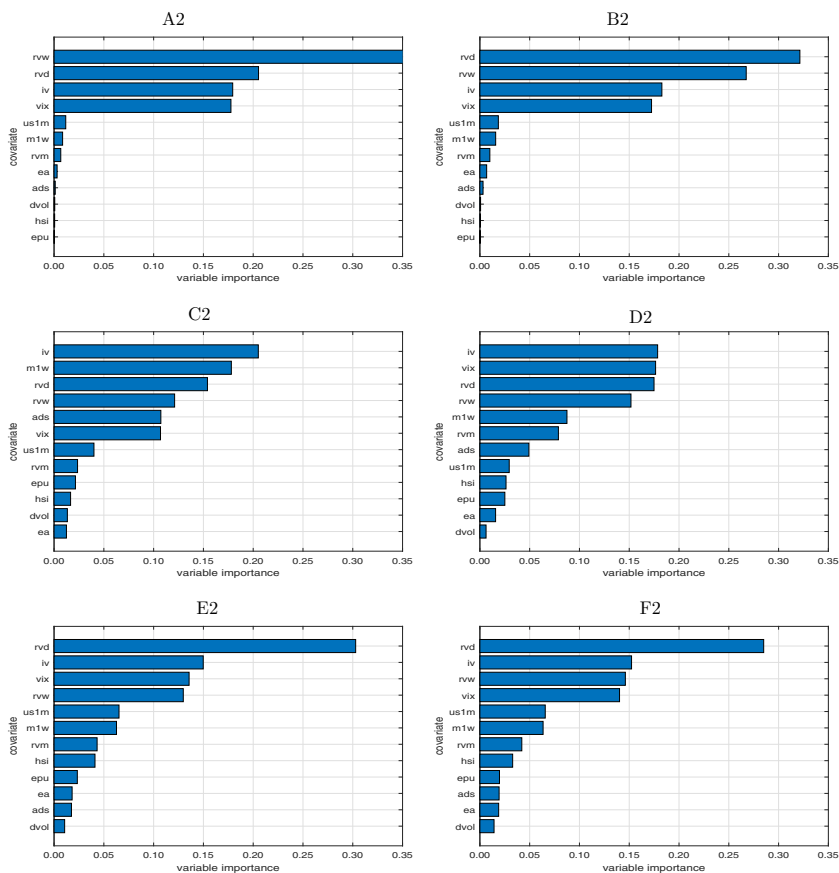
Figure A.2 VI measure. (**A**) A-LA. (**B**) P-LA. (**C**) BG. (**D**) GB. (**E**) NN_1^{10}. (**F**) NN_4^{10}.

Notes: We report the VI measure for each feature of A-LA, P-LA, BG, NN_1^{10}, and NN_4^{10} (NN_3^{10} is omitted to conserve space), sorted according to the numeric value of VI. Refer to the main text for further details.

A.3 In-Sample Parameter Estimates of HAR and HAR-X

Table A.5 Parameter estimates of the HAR and HAR-X model

	HAR	HAR-X
β_0	$-0.006_{(-0.477)}$	$0.005_{(0.378)}$
β_{RVD}	$0.185_{(2.507)}$	$0.110_{(1.501)}$
β_{RVW}	$0.415_{(3.782)}$	$0.365_{(3.151)}$
β_{RVM}	$0.166_{(2.635)}$	$0.079_{(1.281)}$
β_{IV}		$0.198_{(6.390)}$
β_{EA}		$0.043_{(4.656)}$
β_{EPU}		$0.039_{(1.612)}$
β_{VIX}		$0.003_{(0.088)}$
β_{US1M}		$-0.075_{(-2.001)}$
β_{HSI}		$-0.013_{(-0.508)}$
β_{M1W}		$-0.067_{(-3.398)}$
$\beta_{\$VOL}$		$0.046_{(3.361)}$
β_{ADS}		$-0.022_{(-0.662)}$

Notes: The table reports parameter estimates of the HAR and HAR-X model in the initial training set for the one-day-ahead forecasts of Apple's realized variance. In parentheses underneath the coefficient estimate, we report the *t*-statistic for testing statistical significance $H_0 : \beta = 0$, which are based on White's heteroscedasticity-robust standard errors. All variables are standardized.

A.4 Hyperparameter Tuning

The objective of the hyperparameters is to control the bias–variance tradeoff. The purpose of the majority of the hyperparameters is also to reduce the complexity of the model. The challenge is to do it in such a way that the model is capable of extracting relevant information from the observations and does not fit noise. There is a shortage of theoretical results in the literature on how to optimally select hyperparameters, so no superior strategy exists. In this brief Appendix, we explain how we handle it.

We divide the dataset into a training, validation, and test set. In the training set, the model is estimated with various sets of hyperparameters. In the validation set, the models are compared based on their forecasts, and the optimal value of each hyperparameter is selected.[22] The test set serves as the true out-of-sample evaluation, where no estimation or tuning is conducted.

In the paper, we consider both a fixed and rolling estimation window. The less computational heavy is the fixed window. Here, the original data is split at the beginning, the hyperparameters and weights are estimated once and employed throughout the test set. There is no re-estimation through time. The alternative is the rolling window, which allows for time-varying hyperparameters. Here, the hyperparameters are re-estimated each day with a fixed length of the training and validation set. This approach entails inclusion of more recent observations during estimation, as past data are gradually excluded.

---

22 Standard *k*-fold cross-validation has not been conducted, as it violates the time-series structure in our data.

Table A.6 Tuning of hyperparameters

Regularization	BG	RF	GB	NN

$\lambda \in [10^{-5}, 10^2]$*
$\alpha \in [0, 1]$*
	Min. node size = 5
	Trees = 500
		Min. node size = 5
		Trees = 500
		
		Feature split = $J/3$
			Depth $\in \{1, 2\}$*
			Trees
			  $\in \{50, 100, \ldots, 500\}$*
			Learning rate
			  $\in \{0.01, 0.1\}$*
				Learning rate = 0.001
				Epochs = 500
				
				Patience = 100
				
				Drop-out = 0.8
				Initializer: Glorot
				  normal
				Adam = default
				ensemble $\in \{1, 10\}$

Notes: The table presents the hyperparameters of the various ML algorithms. An asterisk indicates that the hyperparameter is tuned in the validation set. Regularization: We do grid-searching over the interval based on a partition consisting of 1000 points for λ and 10 for α. BG and RF: These are the default parameters based on the original Fortran code from Breiman and Cutler (2004). GB: These are the default parameters based on the implementation in Greenwell et al. (2019), who extended the AdaBoost algorithm of Freund and Schapire (1997). NN: These are the default parameters based on the Adam optimizer proposed in Kingma and Ba (2014). The drop-out selection follows Goodfellow, Bengio, and Courville (2016).

Table A.6 shows which hyperparameters are tuned (marked with an asterisk) and, if so, over which interval, or those that are set equal to the default value suggested by the original author.

The interval for λ implies the unregularized regression is a potential solution. The actual grid-searching is here based on partitioning the natural logarithm of $[10^{-5}, 10^2]$, as this allows for extra refinement with smaller degrees of regularization. The optimal value of λ associated with the minimum validation MSE is practically speaking always an interior point. The lower bound is occasionally binding in the $\mathcal{M}_{\text{ALL}}$ dataset (about 2.5% of the times across all estimations), but otherwise not. Typically, only modest regularization is needed. The upper boundary is never hit.

A.5 Regularization of the Neural Network

Adaptive learning rate: To achieve fast convergence and locate a near-minimum function value, an adaptive learning rate is often imposed. The Adam optimizer, applied in this study, achieves that by shrinking the learning rate toward zero during training.

Drop-out: The idea behind drop-out is to temporarily remove neurons from an underlying layer, which in practice means that the output of that neuron is multiplied by zero with strictly positive probability (see, e.g., Srivastava et al. 2014). The drop-out rate is set to 0.8 following Goodfellow, Bengio, and Courville (2016). The drop-out is applied in the training of the network and all connections from the neurons are retained in the validation and test set.

Early stopping: In early stopping, the goal is not to reach a local minimum of the training error, but to stop training the network when there is no improvement for some epochs in the validation error. To ensure sufficiently high patience, we set the patience to 100.
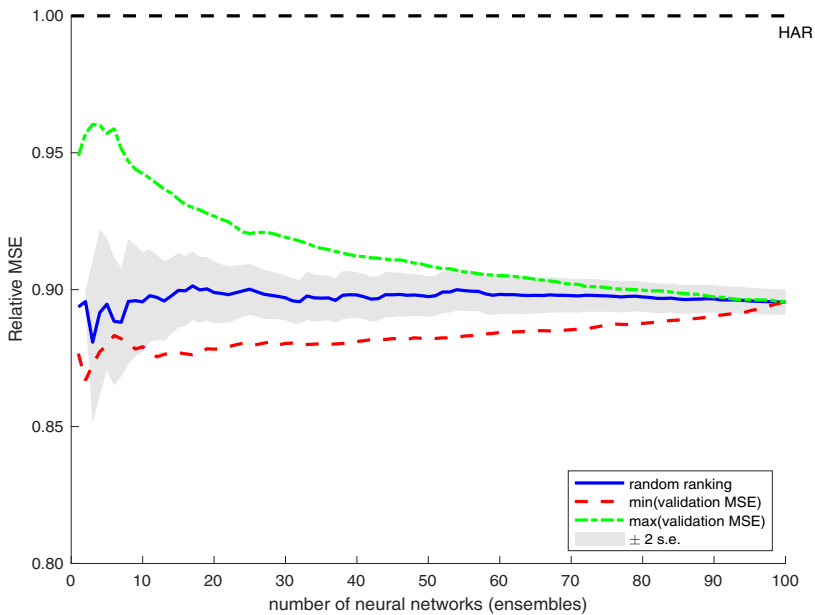
Figure A.3 Robustness check of initial seed for the NN.

Notes: On the *y*-axis, the graph shows the average MSE of NN_2 relative to the HAR model. The *x*-axis denotes the number of NNs (ensembles) out of a hundred that are averaged, when initial seeds are drawn at random (random ranking). The models are also sorted based on highest or lowest MSE in the validation set. The shaded area is ± 2 standard error bands of the random sort. The plot is based on high-frequency data for Apple, but the evolution is representative for other stocks in our sample.

Ensemble: When conducting an ensemble, multiple neural networks are trained, and hereafter the prediction is constructed as an average across the ensemble. When initializing the weights, the Glorot (also called Xavier) normal initializer is chosen. The initialization draws samples from a truncated normal distribution centered on zero.

Initial seed: To ensure our results are not driven by the random initialization, 100 independent neural networks with different seeds are trained and ranked by validation MSE. This allows to reduce variance induced by stochasticity of the Adam optimizer (in the main text, we report the performance of the best, plus an ensemble of the ten best, network from the validation step).

As illustrated in Figure A.3, a fast convergence to the actual performance is seen. Furthermore, changing the initial seed has a negligible impact. To further support this, the figure also shows coverages, when ranking by highest and lowest MSE in the validation set. The relative MSE compared with the HAR model is below one for all initial values. The graph is based on high-frequency data from Apple, but the results are representative for other stocks in our sample.

Glossary

Activation function: A function that takes the weighted sum of inputs (from the previous layer), adds a bias term, and then generates a nonlinear output, which serves as the input for the next layer.

Adam: Adam or "adaptive moments" is an adaptive learning rate optimization algorithm. It can be seen, as a combination of the RMSprop and momentum algorithm with a few distinctions.

Boosting: A ML technique that iteratively combines a set of weak learners to a strong learner.

Epoch: The number of times the algorithm is trained over the entire dataset.

Feature: Feature, explanatory variable, independent variable, and input variables are different terms for the same thing.

Hidden layer: A synthetic layer in a NN. The layer is between the input layer (input features) and the output layer (the prediction).

Hyperparameters: Hyperparameters are higher-level parameters of a model such as how fast it can learn (learning rate) or the complexity of a model.

Input layer: The first layer in the NN, which receives the input features.

Learning rate: The learning rate is the step size of each iteration.

Momentum: A gradient descent based algorithm which helps accelerate the convergence.

Nodes—NN: A node or neuron in a NN takes multiple input values and transforms them into an output value. The transformation is conducted through a nonlinear activation function.

Nodes—Regression tree: *Root nodes*: Entry points to a collection of data. *Inner nodes*: A set of binary questions where each child node is available for every possible answer. *Leaf nodes*: Respond value if reached.

Output layer: The final layer of a NN. The layer containing the predictions.

Regularization: Regularization is a technique used for handling the overfitting problem. By adding a complexity term to the loss function, the sample variance of the model will decrease.

RMSprop: RMSprop (for root mean square propagation) is an adaptive learning rate algorithm. The learning rate is adapted for each of the parameters. RMSprop divides the learning rate by an exponentially decaying average of squared gradients.

Shrinkage: Reducing the effect of sample variation by shrinking the coefficient toward zero.

Sparsity: Sparsity occurs when a large number of elements are set to zero in a vector or matrix.

Subset selection: Selecting a subset of relevant variables.

Test set: A subset of data, at the end of the dataset, to assess the out-of-sample predictive power of the model.

Training set: A subset of data used to fitting/training the parameters in the model.

Validation set: A subset of data used to provide an unbiased evaluation of the current model specification. Often used for tuning the hyperparameters.

Weak learner: A weak learner in a ML model that consistently beats a random guess.

References

Andersen, T. G., and T. Bollerslev. 1998. Answering the Skeptics: Yes, Standard Volatility Models Do Provide Accurate Forecasts. *International Economic Review* 39: 885–905.

Andersen, T. G., T. Bollerslev, and F. X. Diebold. 2007. Roughing It Up: Including Jump Components in the Measurement, Modeling and Forecasting of Return Volatility. *Review of Economics and Statistics* 89: 701–720.

Andersen, T. G., T. Bollerslev, F. X. Diebold, and H. Ebens. 2001. The Distribution of Realized Stock Return Volatility. *Journal of Financial Economics* 61: 43–76.

Apley, D. W., and J. Zhu. 2020. Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 82: 1059–1086.

Aruoba, S. B., F. X. Diebold, and C. Scotti. 2009. Real-Time Measurement of Business Conditions. *Journal of Business & Economic Statistics* 27: 417–427.

Audrino, F., C. Huang, and O. Okhrin. 2019. Flexible HAR Model for Realized Volatility. *Studies in Nonlinear Dynamics and Econometrics* 23: 20170080.

Audrino, F., and S. D. Knaus. 2016. Lassoing the HAR Model: A Model Selection Perspective on Realized Volatility Dynamics. *Econometric Reviews* 35: 1485–1521.

Audrino, F., F. Sigrist, and D. Ballinari. 2020. The Impact of Sentiment and Attention Measures on Stock Market Volatility. *International Journal of Forecasting* 36: 334–357.

Baker, S. R., N. Bloom, and S. J. Nicholas. 2016. Measuring Economic Policy Uncertainty. *The Quarterly Journal of Economics* 131: 1593–1636.

Barndorff-Nielsen, O. E., P. R. Hansen, A. Lunde, and N. Shephard. 2009. Realized Kernels in Practice: Trades and Quotes. *Econometrics Journal* 12: 1–32.

Barndorff-Nielsen, O. E., S. Kinnebrock, and N. Shephard. 2010. "Measuring Downside risk - Realised Semivariance." In T. Bollerslev, J. Russell, and M. Watson (eds.) *Volatility and Time Series Econometrics: Essays in Honor of Robert F. Engle*. Oxford: Oxford University Press, 117–136.

Barndorff-Nielsen, O. E., and N. Shephard. 2002. Econometric Analysis of Realized Volatility and Its Use in Estimating Stochastic Volatility Models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 64: 253–280.

Barone-Adesi, G., F. Bourgoin, and K. Giannopoulos. 1998. Don't Look Back. *Risk* 11: 100–103.

Barone-Adesi, G., K. Giannopoulos, and L. Vosper. 1999. VaR without Correlations for Portfolios of Derivative Securities. *Journal of Futures Markets* 19: 583–602.

Bollerslev, T. 1986. Generalized Autoregressive Conditional Heteroscedasticity. *Journal of Econometrics* 31: 307–327.

Bollerslev, T., J. Li, and Y. Xue. 2018. Volume, Volatility, and Public News Announcements. *The Review of Economic Studies* 85: 2005–2041.

Bollerslev, T., A. J. Patton, and R. Quaedvlieg. 2016. Exploiting the Errors: A Simple Approach for Improved Volatility Forecasting. *Journal of Econometrics* 192: 1–18.

Breiman, L. 1996. Bagging Predictors. *Machine Learning* 24: 123–140.

Breiman, L. 2001. Random Forests. *Machine Learning* 45: 5–32.

Breiman, L, and A. Cutler. 2004. "Random Forests," Fortran Code Version 5.1. Available at https://www.stat.berkeley.edu/breiman/RandomForests/cc_software.htm.

Bucci, A. 2020. Realized Volatility Forecasting with Neural Networks. *Journal of Financial Econometrics* 18: 502–531.

Busch, T., B. J. Christensen, and M. Ø. Nielsen. 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.

Caporin, M., and F. Poli. 2017. Building News Measures from Textual Data and an Application to Volatility Forecasting. *Econometrics* 5: 35–46.

Carr, P., L. Wu, and Z. Zhang. 2020. Using Machine Learning to Predict Realized Variance. *Journal of Investment Management* 18: 1–16.

Chen, L., M. Pelger, and J. Zhu. 2022. Deep Learning in Asset Pricing. *Management Science*, forthcoming.

Chen, T., and C. Guestrin. 2016. "XGBoost: A Scalable Tree Boosting System." In B. Krishnapuram (ed.), *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* New York, NY: Association for Computing Machinery, 785–794.

Christensen, B. J., and N. R. Prabhala. 1998. The Relation between Implied and Realized Volatility. *Journal of Financial Economics* 50: 125–150.

Christensen, K., R. C. A. Oomen, and M. Podolskij. 2014. Fact or Friction: Jumps at Ultra High Frequency. *Journal of Financial Economics* 114: 576–599.

Christensen, K., M. Thyrsgaard, and B. Veliyev. 2019. The Realized Empirical Distribution Function of Stochastic Variance with Application to Goodness-of-Fit Testing. *Journal of Econometrics* 212: 556–583.

Christoffersen, P. F. 1998. Evaluating Interval Forecasts. *International Economic Review* 39: 841–862.

Corsi, F. 2009. A Simple Approximate Long-Memory Model of Realized Volatility. *Journal of Financial Econometrics* 7: 174–196.

Corsi, F., and R. Renò. 2012. Discrete-Time Volatility Forecasting with Persistent Leverage Effect and the Link with Continuous-Time Volatility Modeling. *Journal of Business & Economic Statistics* 30: 368–380.

Cybenko, G. 1989. Approximations by Superpositions of Sigmoidal Functions. *Mathematics of Control, Signals, and Systems* 2: 303–314.

de Prado, M. L. 2018. *Advances in Financial Machine Learning.* 1st edn. Chichester: John Wiley and Sons.

Delbaen, F., and W. Schachermayer. 1994. A General Version of the Fundamental Theorem of Asset Pricing. *Mathematische Annalen* 300: 463–520.

Diebold, F. X., and R. S. Mariano. 1995. Comparing Predictive Accuracy. *Journal of Business & Economic Statistics* 13: 253–263.

Donaldson, R. G., and M. Kamstra. 1997. An Artificial Neural Network-GARCH Model for International Stock Return Volatility. *Journal of Empirical Finance* 4: 17–46.

Engle, R. F. 1982. Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. *Econometrica* 50: 987–1007.

Engle, R. F., E. Ghysels, and B. Sohn. 2013. Stock Market Volatility and Macroeconomic Fundamentals. *Review of Economics and Statistics* 95: 776–797.

Fernandes, M., M. C. Medeiros, and M. Scharth. 2014. Modeling and Predicting the CBOE Market Volatility Index. *Journal of Banking & Finance* 40: 1–10.

Freund, Y. 1995. Boosting a Weak Learning Algorithm by Majority. *Information and Computation* 121: 256–285.

Freund, Y., and R. E. Schapire., 1997. A Decision-Theoretic Generalization of On-line Learning and an Application to Boosting. *Journal of Computer and System Sciences* 55: 119–139.

Friedman, J. H. 2001. Greedy Function Approximation: A Gradient Boosting Machine. *Annals of Statistics* 29: 1189–1232.

Goodfellow, I., Y. Bengio, and A. Courville. 2016. *Deep Learning*. 1st edn. Cambridge, MA: MIT Press.

Greenwell, B. M., B. C. Boehmke, J. P. Cunningham, and G. Developers. 2019. Package "gbm." Preprint arXiv:1805.04755. Available at https://cran.r-project.org/web/packages/gbm/gbm.pdf.

Greenwell, B. M., B. C. Boehmke, and A. J. McCarthy. 2018. "A Simple and Effective Model-Based Variable Importance Measure." Working paper.

Gu, S., B. Kelly, and D. Xiu. 2020. Empirical Asset Pricing via Machine Learning. *The Review of Financial Studies* 33: 2223–2273.

Hansen, P. R., and A. Lunde. 2005. A Forecast Comparison of Volatility Models: Does Anything Beat a GARCH(1,1)? *Journal of Applied Econometrics* 20: 873–889.

Hansen, P. R., A. Lunde, and J. M. Nason. 2011. The Model Confidence Set. *Econometrica* 79: 453–497.

Hillebrand, E., and M. C. Medeiros. 2010. The Benefits of Bagging for Forecast Models of Realized Volatility. *Econometric Reviews* 29: 571–593.

Hoerl, A. E., and R. W. Kennard. 1970. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics* 12: 55–67.

Kingma, D. P, and J. Ba. 2014. Adam: A Method for Stochastic Optimization. Preprint arXiv: 1412.6980.

Koenker, R., and G. Bassett. 1978. Regression Quantiles. *Econometrica* 46: 33–50.

Kupiec, P. H. 1995. Techniques for Verifying the Accuracy of Risk Measurement Models. *The Journal of Derivatives* 3: 73–84.

Lei, Q., X. W. Wang, and Z. Yan. 2020. Volatility Spread and Stock Market Response to Earnings Announcements. *Journal of Banking & Finance* 119: 105126.

Luong, C., and N. Dokuchaev. 2018. Forecasting of Realised Volatility with the Random Forests Algorithm. *Journal of Risk and Financial Management* 11: 61–61.

Maas, A. L., A. Y. Hannun, and A. Y. Ng. 2013. "Rectifier Nonlinearities Improve Neural Network Acoustic Models." In S. Dasgupta and D. McAllester (eds.), *Proceedings of the 30th International Conference on Machine Learning*. New York, NY: Association for Computing Machinery.

Masters, T. 1993. *Practical Neural Network Recipes in C++*, 1st edn. Cambridge, MA: Academic Press.

Mittnik, S., N. Robinzonov, and M. Spindler. 2015. Stock Market Volatility: Identifying Major Drivers and the Nature of Their Impact. *Journal of Banking & Finance* 58: 1–14.

Patton, A. J. and K. Sheppard. 2015. Good Volatility, Bad Volatility: Signed Jumps and the Persistence of Volatility. *Review of Economics and Statistics* 97: 683–697.

Rahimikia, E. and S. H. Poon. 2020. "Machine Learning for Realised Volatility Forecasting." Alliance Manchester Business School, University of Manchester. Working paper.

Schapire, R. E. 1990. The Strength of Weak Learnability. *Machine Learning* 5: 197–227.

Schwert, G. W. 1989. Why Does Stock Market Volatility Change Over Time. *The Journal of Finance* 44: 1115–1153.

Siggaard, M. 2022. "Implied Versus Realized Volatility: Volatility Patterns around Announcements." *Working paper*, Aarhus University.

Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research* 15: 1929–1958.

Taylor, S. J. 1982. "Financial Returns Modeled by the Product of Two Stochastic Processes - a Study of the Daily Sugar Prices 1961–75," In O. D. Anderson (ed.), *Time Series Analysis: Theory and Practice*. Amsterdam: North-Holland, 203–226.

Tibshirani, R. 1996. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society: Series B (Methodological)* 58: 267–288.

Varian, H. R. 2014. Big Data: New Tricks for Econometrics. *Journal of Economic Perspectives* 28: 3–28.

Zou, H. 2006. The Adaptive Lasso and Its Oracle Properties. *Journal of the American Statistical Association* 101: 1418–1429.

Zou, H., and T. Hastie. 2005. Regularization and Variable Selection via the Elastic Net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67: 301–320.