# Automated Volatility Forecasting*

Sophia Zhengzi Li† and Yushan Tang‡

First Draft: November 18, 2020

This Version: May 19, 2023

**Abstract**

We develop an automated system to forecast volatility by leveraging over one hundred features and five machine learning algorithms. Considering the universe of S&P 100 stocks, our system results in superior out-of-sample volatility forecasts compared to existing risk models across forecast horizons. We further demonstrate that our system remains robust to different specifications and is scalable to a broader S&P 500 stock universe via hyperparameter transfer learning. Finally, the statistical improvement in volatility forecasts translates into an enhanced annual return around 8.5% from a cross-sectional variance risk premium strategy.

**JEL Classification**: C13, C14, C52, C53, C55, C58.

**Keywords**: Automation; Machine Learning; Volatility Forecasting; High-Frequency Data; Transfer Learning.

*A previous version of the paper was circulated under the title "Forecasting realized volatility: An automatic system using many features and many machine learning algorithms". We thank Torben Andersen, Tim Bollerslev, Will Cong, Bjørn Eraker, Todd Griffith (discussant), Yufeng Han, Zhongzhi He (discussant), Hao Jiang, Yuan Liao, Zheng Long (discussant), Markus Pelger, Andrea Tamoni, Allan Timmermann, Yuanyuan Xiao, Dacheng Xiu, Peixuan Yuan, Aurelio Vasquez (discussant), Guofu Zhou, and seminar participants at Rutgers Business School, Johns Hopkins Carey Business School, Michigan State Broad College of Business, UNC Charlotte Belk College of Business, Texas A&M Mays Business School, Southwestern University of Finance and Economics, University of Science and Technology of China, Nankai Business School, NBER-NSF Time Series Conference, FMA Annual Meeting, FMA Conference on Derivatives and Volatility, Global AI Finance Research Conference, Greater China Area Finance Conference, University of Florida Research Conference on Machine Learning in Finance, and Chinese Finance Annual Meeting for their helpful comments and suggestions.

†Rutgers Business School, 1 Washington Park, Newark, NJ 07102; E-mail: zhengzi.li@business.rutgers.edu.

‡Rutgers Business School, 1 Washington Park, Newark, NJ 07102; E-mail: yushan.tang@rutgers.edu.

# 1. Introduction

In the realm of risk management and asset pricing, volatility plays a critical role. In risk management, volatility is a key input of almost all risk models that are essential to investment or regulatory decision makings. In asset pricing, volatility directly impacts the price of derivatives and arguably is related to the expected return of stocks. The availability of high-frequency price data over the past two decades has spurred the field of modeling and forecasting realized variance, $RV$, which is an accurate measure of volatility.[1] Most of the existing $RV$ forecasting models rely on a handful of predictors and utilize them one by one within the framework of classical statistical inference. In this paper, instead of arguing the dominance of a particular feature or algorithm, we have an ambitious objective - building an *automated* forecasting system specially tailored to *volatility* prediction that: 1) reduces human intervention in choosing features and algorithms; 2) scales to fit many features while controlling for overfitting; 3) utilizes more flexible and state-of-the-art learning algorithms; and 4) achieves good and consistent out-of-sample performance.

To achieve these objectives, our system is designed with several distinct elements. First, the system is inclusive of predictors. Instead of focusing on a few well-engineered features, we consider many potentially useful features altogether and let the fitters decide how to combine them automatically. The intuition is that a large and diverse set of features will have the benefit of diversification and perform better out-of-sample. Our recommended feature sets include a range of well-known $RV$-based features as well as the implied-volatility surface that receives far less attention in the volatility forecast literature perhaps due to the challenges posed by the large parameter dimension.[2] Second, we apply a wide range of learning algorithms beyond the traditional OLS models for capturing different types of predictive information from the large feature sets. At the next level, we consider ensemble methods for combining results from those learning algorithms to achieve stable out-of-sample performance.

We show the effectiveness of our automated forecasting system through the *largest-scale*

[1] Andersen and Bollerslev (1998) propose the use of realized volatility for accurately measuring the true latent integrated volatility. Recently, Da and Xiu (2021) develop a simple estimator of volatility using high-frequency data in the presence of noises.

[2] The main results use 16 well-constructed $RV$-based features from the literature. However, we find that replacing them by the simple raw lagged daily $RV$'s can yield comparable volatility forecast under our system in Section 5.4.

experiment in the volatility forecasting literature that compares different combinations of features and learning algorithms for 173 stocks that were S&P 100 Index constituents and another 663 stocks that were S&P 500 Index constituents spanning more than two decades. Our automated system is able to deliver superior out-of-sample forecasting performance, which can further translate into significant gains for a cross-sectional variance risk premium strategy.

We start by comparing the predictive power of different features for the S&P 100 stock sample using the traditional OLS fit. We consider 16 realized-variance-based ($RV$-based) features from five popular volatility forecasting models including the HAR model by Corsi (2009), the MIDAS model by Ghysels, Santa-Clara, and Valkanov (2006), the SHAR model by Patton and Sheppard (2015), the HARQ-F model by Bollerslev, Patton, and Quaedvlieg (2016b), and the HExpGl model by Bollerslev, Hood, Huss, and Pedersen (2018), as well as 102 implied-variance-based ($IV$-based) features across all deltas and with maturity between one and three months. We find that the forecasting performance of any stand-alone $RV$-based feature set can be improved if we combine them all together, whose performance can be further enhanced by adding the $IV$-based features.[3]

After fixing the feature set, we evaluate the performance of five learning algorithms, including LASSO, Principal Component Regression (PCR), Random Forecast (RF), Gradient Boosted Regression Trees (GBRT), and Neural Network (NN). We find that machine learning algorithms can improve performance over that of OLS models. Further, a simple average ensemble model that combines all machine learning algorithms delivers even more superior performance across forecast horizons, with relative out-of-sample $R^2$'s ($R^2_{OOS}$'s) equal to 9.0%, 14.3%, 15.2%, and 10.0% at daily, weekly, monthly, and quarterly horizons. The corresponding relative $R^2_{OOS}$'s jump to 10.4%, 18.7%, 29.6%, and 27.5% for the most recent decade, indicating that our automated volatility forecasting system becomes increasingly powerful over time.

To enhance the interpretability of forecasts from our automated system, we investigate the temporal dependence structure and feature group importance inherent in our system. Recognizing that the complexity of our system may hinder economic insights, we employ two approaches. First, we project our model forecasts onto a linear space spanned by lagged daily $RV$'s to understand the dynamics of the model forecasts. Additionally, we introduce a permutation group importance

[3] Eraker (2004) illustrates the advantages of using both options and stock data to examine the empirical performance of jump diffusion models of stock price dynamics. Han, Liu, and Tang (2020) show that option prices can predict downward jumps in stock prices.

measure to assess the contributions of feature groups within machine learning models. Both approaches serve to unravel the workings of our black-box system and add to our understanding of how it achieves superior out-of-sample performance. In terms of temporal dependence, we find the implied dependency structure of our system differs from existing models. Our system also dynamically weighs past information according to the forecast horizon, potentially contributing to its superior out-of-sample performance. With respect to feature group importance, we observe that all three groups of features contribute at least 10% to the forecast across most horizons and learning algorithms. Interestingly, implied variance features increasingly contribute more to volatility forecasting over time. We posit that much of the gain is attributable to the improved quality of implied variance features as the overall option market becomes more liquid.

We perform several robustness tests. First, we consider three alternative NN model specifications and find that the performance of NN is not particularly sensitive to the NN architecture. Second, we consider alternative ensemble methods with various weighting schemes, including data-driven dynamic weighting. We find that none of these ensemble methods can dominate the simple average method, which requires the least human intervention. Third, we add new features of firm characteristics and pure noises to the model. We find that firm characteristics do not significantly improve the OOS performance and they are only marginally important as a group in group importance evaluation and the pure noise features do not hurt the OOS performance and are almost of zero importance in group importance evaluation. Fourth, to further challenge our system, we replace the well-engineered RV features in the literature with raw lagged $RV$'s. The performance of our system based on these raw inputs is comparable with those based on engineered features, suggesting that the system is powerful for extracting volatility signals.

Is our machine-learning-based automated system scalable to more stocks? To this end, we examine its performance on a large and different set of 663 S&P 500 stocks. To speed up hyperparameter tuning for nonlinear models, we directly transfer tuning parameters for RF and GBRT learned from the S&P 100 stock universe to the S&P 500 universe, and retrain both models without validating these tuning parameters. The remaining ML-based and OLS-based models can be estimated efficiently as the sample grows and are completely recalibrated using the S&P 500 universe. We find that tuning parameters based on the original sample perform well in the new sample, and our automated system consistently delivers significant gains over the traditional OLS-based approach.

To more concretely evaluate the economic significance of our automated system, we examine a volatility-based trading strategy that relies on the cross-sectional predictability of the variance risk premium, as motivated by Han and Zhou (2011). Our results demonstrate that the superior out-of-sample prediction performance from our system translates into an enhanced annualized return of around 8.5% for a long-short portfolio sorted by the variance risk premium. This return spread remains significant even after accounting for exposures to common risk factors.

Our paper makes two main contributions to the literature, both in methodology and in new empirical findings. Regarding methodology, we propose a modern machine-learning-based framework specifically designed for volatility forecasting, which consists of feature engineering and learning algorithm fitting steps. In the feature engineering step, we consider many features *all* together, and use learning algorithms along with prediction-oriented model selection procedures to *automatically* and *dynamically* select features. In the learning algorithm fitting step, we illustrate the importance of algorithm fitting using panel data instead of time series data, and go beyond traditional OLS widely used in $RV$ forecasting to include both linear and nonlinear learning algorithms. We do not argue for the dominance of one particular algorithm over another as suggested by Wolpert (1996), but consider combinations across all learning algorithms as long as they are well implemented to avoid overfitting. As a result, our framework is less prone to human decision-making biases (e.g., cherry-picking of features and models) and interventions (e.g., using one set of features or models for a particular sample period) and appears to be robust throughout our analyses.

Regarding new empirical findings, we conduct the largest-scale experiment involving the forecasting of stock realized volatility. Our big dataset consists of intraday high-frequency data and stock-level option data for 173 S&P 100 stocks and another 663 S&P 500 stocks over the period from January 1996 to June 2019. Our giant feature set includes predictors drawn from five popular $RV$-based volatility forecasting models and implied variances with one-to-three-month maturity across all deltas. Our learning algorithms consist of major linear and nonlinear machine learning models. With our comprehensive data and unique study design, we empirically demonstrate the gains that can be obtained using many features and learning algorithms via our automated system to forecast realized volatility.

Our paper adds to the growing literature on applying machine learning (ML) techniques to predictive problems in asset pricing. This literature shows the power of ML in predicting stock

returns, corporate bond returns, and mutual fund returns.[4] Unlike these studies mostly focusing on return prediction, we focus on volatility forecast which presents a unique set of opportunities and challenges. First, volatility is known to be persistent, meaning that it is a high signal-to-noise ratio problem as opposed to the low signal-to-noise ratio problem of return prediction (e.g., an $R^2$ of 50% or higher versus 10% or lower). This makes volatility forecast suitable for applying sophisticated machine learning algorithms, which are mostly developed for improving performance in high signal-to-noise ratio scenarios such as image classification and recommendation system (e.g., error rate 6% – 60%).[5] On the other hand, the high signal-to-noise ratio raises the bar for our system to beat. To improve performance, we exploit volatility-specific properties such as the implied-volatility surface for improving features and commonality among stocks for improving fitting. Second, returns are known to be predicted by many non-market data features such as firm characteristics. In contrast, few conclusions are reached about the relationship between non-market data and future volatility. Based on the new framework, we offer some comprehensive evidence of the predictive power of non-market data for volatility forecast.

In addition, several papers apply selective ML algorithms to volatility forecasting problems: Audrino and Knaus (2016) use LASSO to forecast realized volatility; Luong and Dokuchaev (2018) forecast realized volatility with random forest algorithms; Bucci (2020) and Rahimikia and Poon (2023) apply neural networks to predict realized volatility; and Carr, Wu, and Zhang (2020) rely on Ridge, Feedforward Neural Networks, and Random Forecast to predict realized variance of SPX. Compared to studies that apply machine learning to volatility forecasting, our work focuses on building an entire learning system with the most comprehensive feature set and algorithms that is automated and robust for a much broader stock universe. Again, we emphasize the benefits of using not just one or two particular learning algorithms but more specifically the benefits of an ML-based automated volatility system that allows us to consider features and algorithms more inclusively, because machines are able to scan, fit, and select features in a robust and prediction-error-optimized fashion.

The paper is organized as follows. Section 2 discusses the data and features used in the paper.

[4] See, e.g., Rapach, Strauss, and Zhou (2013), which is the first to use LASSO in finance in modeling returns; Gu, Kelly, and Xiu (2020) and Chen, Pelger, and Zhu (2023) on predicting stock returns; Bali, Goyal, Huang, Jiang, and Wen (2022) on predicting corporate bond returns; and Li and Rossi (2021) and Kaniel, Lin, Pelger, and Van Nieuwerburgh (2022) on predicting mutual fund returns.

[5] For examples of error rates in image classification and recommendation system, see Table 8 in Russakovsky et al. (2015) and Figure 6 in Bao and Jiang (2016).

Section 3 summarizes the details of the machine learning methodology and evaluation metrics. Section 4 presents the out-of-sample performance of various forecasting models. Section 5 details the robustness studies. Section 6 examines the out-of-sample performance of our system on a broader set of S&P 500 stocks. Section 7 demonstrates the economic gains of our automated forecasting system. Section 8 concludes. Further details regarding the data cleaning rules, feature construction, algorithm fitting, and additional results are provided in the Appendix.

## 2. Data and Variables

### *2.1. Data*

We consider a large universe of stocks that were ever constituents of the S&P 100 index over the period from January 1993 to June 2019, listed on the New York Stock Exchange (NYSE), National Association of Securities Dealers Automated Quotations (NASDAQ), and the American Stock Exchange (AMEX) with share code of 10 or 11, price between \$1 and \$1,000, and daily number of trades greater than or equal to 100. To prepare the intraday price data, we collect minute-by-minute observations of intraday prices from the NYSE trade and quote (TAQ) database by applying the cleaning rules of Bollerslev, Li, and Todorov (2016a), Bollerslev, Li, and Zhao (2020), and Jiang, Li, and Wang (2021). The data cleaning rules are detailed in Section A.1 of the Appendix. In addition to the TAQ data, we collect implied variances for the same universe of stocks from the volatility surface data in OptionMetrics. The database provides implied volatilities with various maturities and deltas at the stock and date levels. In our empirical analyses, we rely on implied variances (i.e., squared implied volatilities) from call and put options with maturity of one month (30 days), two months (60 days), and three months (91 days), and absolute delta of 0.1, 0.15, ..., 0.9.[6] Some $RV$-based features (e.g., features from the HExpGl model) require a longer historical sample for estimation. To ensure that all $RV$-based features have the same history, we use the sample between 1993 and 1995 to construct their first observations; therefore our features first become available in January 1996. Our final S&P 100 stock sample consists of 173 unique stocks with at least five years of data for all features and response variables over the period from January 1996 to June 2019. On average, there are 138 unique stocks per day.

[6]Implied variances with ten-day maturity only became available in November 2005 for a handful of stocks and are excluded from our analyses because of limited availability of data.

Our main sample of the S&P 100 stock universe is large-scale by the volatility forecasting literature standard.[7] The focus on S&P 100 stocks helps ensure all stocks are frequently traded and thus their realized features based on intraday data are less subject to measurement errors. In subsequent sections, we further examine the out-of-sample performance of various models for the 663 S&P 500 stocks that are not S&P 100 constituents via transfer learning, and then the economic gain for all 836 S&P 500 stocks.[8] As far as we know, our universes are the largest ever that has been explored in the $RV$ forecasting literature.

### *2.2. Response Variable*

In this paper, we aim to predict realized variance ($RV$), which is a consistent estimator of the quadratic variation of the log price process over a given period. Formally, let $p_{i,t}$ denote the natural logarithm of stock $i$'s price on day $t$. We omit subscription $i$ in this section for simplicity and assume the log price follows a generic jump diffusion process:

$$p_t = \int_0^t \mu_s ds + \int_0^t \sigma_s dW_s + J_t, \tag{1}$$

where $\mu_t$ and $\sigma_t$ denote the drift and diffusive volatility processes, respectively, $W$ is a standard Brownian motion, $J$ is a pure jump process, and the unit time interval corresponds to a trading day. It is natural to extend the notation to intraday prices using the notation $p_t, p_{t+1/n}, ..., p_{t+1}$, assuming prices are observed at $n+1$ equally spaced time intervals from day $t$ to day $t+1$. The *annualized* daily $RV$ based on summing over frequently sampled squared returns within a trading day is then:

$$RV_t^d = 252 \times \sum_{i=1}^{n} r_{t-1+i/n}^2, \tag{2}$$

where $r_{t-1+i/n} = p_{t-1+i/n} - p_{t-1+(i-1)/n}$ is the log return over the $i$th time interval on day $t$. In particular, we include the overnight squared returns in the daily $RV$ estimation to obtain an

[7] Another paper we are aware of that uses such a large dataset to forecast volatility is Patton and Sheppard (2015), which relies on 105 unique stocks that were constituents of the S&P 100 index and with four-year continuous data between June 1997 and July 2008.

[8] To demonstrate that the superior performance of our system remains intact for any potential selection biases, we carry out supplementary evaluations. These evaluations involve relaxing the requirement of five-year data for all features and response variables, and excluding sample period before a stock is formally included in the S&P 500 index, as detailed in Section A.4.5 in the Appendix.

$RV$ measure for the entire day. As shown in Andersen, Bollerslev, Diebold, and Labys (2001, 2003), $RV$ is a consistent estimator for quadratic variation when the number of intervals $n \to \infty$. Longer-horizon $RV$'s (e.g., weekly, monthly, and quarterly) can be estimated by averaging daily $RV$ over the corresponding intervals. Formally, the $h$-day ahead $RV$ is defined as:

$$RV_{t+1}^{t+h} = \frac{1}{h} \sum_{i=1}^{h} RV_{t+i}^{d}, \tag{3}$$

where $h = 5,\ 21,\ 63$ corresponds to weekly, monthly, and quarterly $RV$, respectively.[9]

Our research objective is to build better predictive models for the responses of daily, weekly, monthly, and quarterly $RV$'s. To empirically compute $RV$, we use the five-minute sampling frequency commonly employed in the realized volatility literature.[10] To further increase the efficiency of $RV$ estimates, we apply a subsampling approach following Zhang, Mykland, and Aït-Sahalia (2005). Specifically, we compute five separate daily $RV$ estimates by starting the trading day at 9:30, 9:31, 9:32, 9:33, and 9:34, respectively, and then average over these five estimates to obtain the final daily $RV$ estimate.

### *2.3. Features*

Our research design involves first constructing input features that potentially contain predictive information, then fitting learning algorithms to estimate functions that map features to the response variable, and finally evaluating the performance of our predictions. We consider two types of features: 1) realized features proposed by popular $RV$-based volatility forecasting models: HAR, MIDAS, SHAR, HARQ-F, and HExpGl; and 2) implied variance features.[11] The table below summarizes the realized features from each risk model along with the option-implied features from OptionMetrics. Definitions of the features are detailed in Section A.2 of the Appendix.

Table 1 reports pairwise correlations for all realized features and selected implied variance

[9] Volatilities are shown to exhibit horizon effects. For example, Carvalho, Lopes, and McCulloch (2018) find that under plausible prior specifications, stocks are less volatile in the long run, whereas Kamara, Korajczyk, Lou, and Sadka (2016) show that systematic factors that are portfolio excess returns tend to exhibit volatility at longer horizons greater than a proportionate scaling up of short-horizon volatility.

[10] Liu, Patton, and Sheppard (2015) compare more than 400 different $RV$ estimators across multiple asset classes and conclude that it is difficult to significantly beat the 5-minute $RV$.

[11] Christensen and Prabhala (1998) find that volatility implied by S&P 100 index option prices predicts ex-post realized volatility. Busch, Christensen, and Nielsen (2011) further show that implied volatility contains incremental information about future volatility across different asset classes.

| Source | Features |
|---|---|
| HAR | $RV^d$, $RV^w$, $RV^m$, $RV^q$ |
| MIDAS | $MIDAS$ term for the corresponding forecast horizon |
| SHAR | $RVP^d$, $RVN^d$, $RV^w$, $RV^m$, $RV^q$ |
| HARQ-F | $RV^d$, $RV^w$, $RV^m$, $RV^q$,<br>$RV^d\sqrt{RQ^d}$, $RV^w\sqrt{RQ^w}$, $RV^m\sqrt{RQ^m}$, $RV^q\sqrt{RQ^q}$ |
| HExpGl | $ExpRV^1$, $ExpRV^5$, $ExpRV^{25}$, $ExpRV^{125}$, $ExpGlRV$<br>(# of unique realized features from all five models = 16) |
| OptionMetrics | $CIV^{jm,\delta}$ and $PIV^{jm,-\delta}$, $j = 1, 2, 3$, $\delta = 0.1, 0.15, ..., 0.9$<br>(# of implied variance features = 102) |

features with absolute delta equal to 0.5.[12] MIDAS features for various forecast horizons exhibit the highest correlations of 0.96 or above with each other likely because they are calibrated by fitting highly correlated dependent variables to the same daily $RV$ terms as shown in Eq. (A.3) in the Appendix. For a given forecasting horizon, we only use one MIDAS feature corresponding to that particular horizon so that the high correlations among MIDAS features will not cause multicollinearity. HARQ-F features (e.g., $RV^k\sqrt{RQ^k}$) have weak correlations with other realized features, mostly because these features contain realized quarticities while other realized measures are all linear combinations of daily $RV$'s. Interestingly, $IV$-based features $CIV$s and $PIV$s exhibit relatively weak correlations with all $RV$-based features, suggesting potentially new information contained in the $IV$-based features to the $RV$-based features.

## 3. Machine Learning Methodology

We investigate five machine learning (ML) algorithms in total. The first two are linear: Least Absolute Shrinkage and Selection Operator (LASSO) and Principal Component Regression (PCR). The next three are nonlinear: Random Forest (RF), Gradient Boosted Regression Trees (GBRT), and Neural Network (NN). In this section, we first discuss the training and validation scheme of each model, and then explain how we standardize features in certain models. Next, we introduce an ensemble model based on the five individual algorithms. Lastly, we propose a new group importance

[12] Table A.1 in the Appendix provides the steps to construct the final stock sample and the descriptive statistics of the features.

metric. Detailed descriptions of these ML algorithms are provided in Section A.3 of the Appendix.

### *3.1. Training and Validation*

Machine learning algorithms include key hyperparameters that control for model complexity. We should tune these parameters based on the *prediction* error rather than the *training* error. Otherwise, learning algorithms, especially nonlinear algorithms, will overfit the training data and do poorly out of sample. Accordingly, we adopt a training-validation-testing scheme for hyperparameter calibration and model assessment. As discussed later in Section 4.4, we purposely fit pooled models based on panel data in order to increase estimation efficiency over stock-by-stock fitting. Specifically, at the end of each year $t$, we divide the sample into three parts: an expanding-window training set consisting of data from data inception in year 1996 to year $t-1$ (a minimum of four years), a validation set consisting of year $t$ data, and a testing set consisting of year $t+1$ data. In other words, we refit our models every year by increasing the training set by one year, and rolling the validation and testing sets one year forward.

Our training-validation-testing scheme allows us to estimate a model with 118 predictors using data at a much higher dimension (i.e., $N \gg P$). For example, our first training sample contains four years of data from year 1996 to 1999, and our last training sample contains 22 years of data from year 1996 to 2017. Given there are 138 unique stocks per day on average, our first training sample includes more than 130,000 observations ($138 \times 252 \times 4$), and our last training sample includes more than 760,000 observations ($138 \times 252 \times 22$).

Our scheme leaves us with a total of 19 years of predictions between 2001 and 2019 corresponding to 19 fitted models for each learning algorithm. For models that do not require validation (e.g., OLS), we use data from data inception to year $t$ for training and data in year $t+1$ for testing. Thus, the overall testing sets are the same across models and differences in model performance cannot be driven by sample differences.

### *3.2. Feature Standardization*

Before fitting LASSO, PCR, and NN, we standardize the features in each training-validation-testing sample using the training sample mean and standard deviation. We perform feature standardization on these models because: 1) by design, LASSO shrinks large coefficients towards zero via regularization,

which in turn introduces scale sensitivity into model estimation; 2) PCR first performs PCA on the feature set to construct principal components, and PCA can be sensitive to the variances of the initial features; and 3) NN with flexible activation functions in hidden layers requires significant resources for numerical computation, and unscaled features can lead to slower convergence and increase the likelihood of sticking in local optima. Other than LASSO, PCR, and NN, standard OLS and tree-based models are scale equivariant and do not require feature standardization.

### *3.3. Ensemble Model*

For a given forecast horizon, we denote the $RV$ prediction from ML algorithm $m$ by $\widehat{g}^m(z_{it};\theta^m)$, where $z_{i,t}$ is the feature vector for stock $i$ on day $t$ and $\theta^m$ is the unknown model parameter. After obtaining forecasts from each stand-alone learning algorithm, we consider an ensemble model that combines forecasts from several models. The intuition is that no single model is expected to dominate the others under any circumstances (Wolpert, 1996). Different models might do well in different scenarios and by combining them we can make the forecast more robust. Our main results are based on a simple equal-weighted average of all five machine learning methods as our ensemble forecast and call it AVG:

$$AVG = \frac{1}{5}\sum_{m=1}^{5}\widehat{g}^m(z_{i,t};\theta^m). \tag{4}$$

In the robustness analysis of Section 5.2, we study different weight schemes including model-driven dynamic weighting.

### *3.4. Performance Evaluation*

Since we focus on prediction rather than statistical inference, we use out-of-sample $R^2$ relative to a benchmark as our main performance measure:

$$R^2_{OOS}(m) = 1 - \frac{\sum_{i,t}(RV_{i,t} - \widehat{RV_{i,t}}^{m})^2}{\sum_{i,t}(RV_{i,t} - \widehat{RV_{i,t}}^{benchmark})^2}, \tag{5}$$

where $\widehat{RV_{i,t}}^{m}$ refers to forecasts from one of the OLS-based or machine-learning-based volatility forecasting models, and $\widehat{RV_{i,t}}^{benchmark}$ is the forecast of a benchmark model.[13] A positive $R^2_{OOS}(m)$ indicates that model $m$ achieves smaller out-of-sample prediction mean squared errors than the benchmark model. We consider two benchmarks: one is the prediction from HAR, and the other is the long-run mean, which equals the expanding sample mean of $RV$'s from the inception date until day $t$. The long-run mean is a commonly used benchmark and also mirrors the out-of-sample evaluation measure used in the return prediction literature. However, the bar for beating the long-run mean is low because volatilities are persistent and time-varying. HAR is perhaps a better benchmark because it empirically shows good volatility forecasting performance and is also easily implementable and interpretable.

In addition to $R^2_{OOS}(m)$, we also use a modified Diebold and Mariano (1995) (DM) test for pairwise comparison of two models. The DM test is based on the difference in the out-of-sample squared error losses between two forecasting models. More formally, for stock $i$ on day $t$, the loss differential is defined as $d_{i,t} = (\widehat{e}^{(1)}_{i,t})^2 - (\widehat{e}^{(2)}_{i,t})^2$, where $\widehat{e}^{(1)}_{i,t}$ and $\widehat{e}^{(2)}_{i,t}$ are the prediction errors from two models. We then compute the cross-sectional mean of $d_{i,t}$ and denote it by $d_t$. The modified DM test statistic $DM = \overline{d}/\widehat{\sigma}_d$, where $\overline{d}$ and $\widehat{\sigma}_d$ are the mean and Newey and West (1987) standard error of $d_t$ over the testing sample.

### *3.5. Group Importance Metric*

To shed additional light on how these features and learning algorithms work for volatility forecasting, we investigate how different features contribute to the prediction at various horizons. Our feature importance evaluation process differs from those in existing applications of machine learning to asset pricing. For example, Gu, Kelly, and Xiu (2020) compute the reduction in $R^2$ obtained by setting all values of one feature $j$ to zero within each training set and then averaging the reductions over the training samples to obtain a per-feature importance measure. In our evaluation, we consider per-group feature importance instead of per-feature variable importance by assigning highly correlated features to one group and computing the importance of the entire group. The motivation is to avoid the dilution effect of per-feature variable importance. Consider the following simple

[13] Mirroring Swanson and White (1997) and Bollerslev, Hood, Huss, and Pedersen (2018), we apply an "insanity filter" to avoid deflation in $R^2_{OOS}$. Specifically, we replace any predictions that exceed (fall below) the maximum (minimum) outcome value in the training sample with the observed maximum (minimum).

example. Suppose two uncorrelated features $X_1$ and $X_2$ are equally important so they reduce $R^2$ equally by 0.5 from the joint model $(X_1, X_2)$. Now suppose a new feature $X_3$ is introduced and $X_3$ is highly correlated with $X_2$ but not with $X_1$. If the model is estimated in a sensible way, then the variable importance of $X_2$ measured by its marginal reduction of $R^2$ will be diluted by $X_3$ because $X_3$ might serve as a proxy for $X_2$ in the model. The dilution phenomenon would be more pronounced when there are many more correlated variables. For our feature set, several subgroups are highly correlated as shown in Table 1. Because of the dilution effect, the per-feature variable importance measure might not truly reflect the importance of that feature. Therefore, we consider per-group feature importance to reduce cross-group correlations and also to reduce the number of candidates for feature importance evaluation.

Second, instead of setting all values of the features within a group to zero, we consider random permutations of values across observations within the training set for the tested features as suggested by Fisher, Rudin, and Dominici (2019). This is because setting all feature values to zero introduces unintended bias to nonlinear models. To offer a simple example, suppose that we wish to test the marginal contribution of daily realized variance $RV_t^d$ in RF fitting. If we simply set $RV_t^d$ to zero, all observations will fall into one child node at each binary split that uses $RV_t^d$ as the splitting variable, causing severe bias in the prediction. In contrast, permutation breaks the association between features and the true outcome, enabling us to remove the effects of the tested features. Specifically, for each training set and each feature group $k$, we permutate all values of each feature within group $k$ using the panel of stocks without replacement and record the corresponding $R^2$. To reduce the permutation variance, we repeat the permutation five times and use average $R^2$ to compute the reduction in $R^2$. We then average the reductions in $R^2$ over different training samples to obtain a single group importance measure $GI_k$.

## 4. Out-of-Sample Performance of Forecasting Models

In this section, we show how machine learning can improve the volatility forecasting performance over that of traditional approaches. We begin by establishing the baseline performance by applying the traditional OLS method to *each* of the feature sets described in Section 2.3, as is commonly done in the literature. We then show that combining many $RV$-based features improves performance

over that of any stand-alone feature set and that including the new $IV$-based features adds further value to $RV$-based features. After fixing the feature set, we demonstrate the benefits of more sophisticated learning algorithms in comparison with the baseline OLS, and show that an ensemble model combining many learning algorithms delivers superior performance across all forecast horizons and time periods. Furthermore, we illustrate the benefits of using panel data in model fitting and provide insights into the temporal dependency structure inferred by the models. Lastly, we evaluate the importance of different feature groups.

### *4.1. OLS-Based Models*

In Table 2 we report the out-of-sample performance of OLS-based volatility forecasting models based on the $R^2_{OOS}$ relative to HAR from Eq. (5).[14] The first column lists the model names and the second column summarizes their features. First, we focus on the four popular $RV$-based models. Among them, MIDAS, SHAR, and HARQ-F outperform HAR across all forecast horizons, as is evident by the positive relative $R^2_{OOS}$'s. HExpGl outperforms HAR at the daily, weekly, and monthly horizons, while slightly underperforms HAR at the quarterly horizon.

Next, we combine all 16 realized measures from the MIDAS, SHAR, HARQ-F, and HExpGl models through OLS.[15] This model, namely $OLS^{RM}$, not only outperforms HAR by wide margins across all horizons, but it also generally beats individual models in most cases. Only HARQ-F has a higher relative $R^2_{OOS}$ than $OLS^{RM}$ at the quarterly horizon. Overall, the superior performance of $OLS^{RM}$ illuminates the importance of feature combination in improving volatility forecasting performance.

We then fit OLS to the 102 implied variances ($IV$'s) from call and put options with one-, two-, and three-month maturities and denote the model by $OLS^{IV}$. Unlike the realized features, these $IV$ features seem to underperform HAR as measured by the relative $R^2_{OOS}$'s. However, this does not mean that $IV$ features are useless in the presence of realized features. Although $IV$'s are weakly informative as stand-alone features, they can still add value as long as they contain information that is orthogonal to the realized features. To test whether there is any additional value gained from $IV$ features, we expand the feature set in $OLS^{RM}$ by adding the 102 $IV$ features to the 16

[14] $R^2_{OOS}$'s relative to the long-run mean of $RV$'s for OLS-based models are presented in Table A.2 in the Appendix.

[15] For a given forecast horizon, we include only one $MIDAS$ term corresponding to the same horizon. For instance, in predicting weekly $RV$, we keep the $MIDAS$ term constructed by using coefficients estimated from forecasting weekly $RV$ according to Eqs. (A.2) and (A.3) in the Appendix.

realized features and call the model $OLS^{ALL}$. The row labeled $OLS^{ALL}$ reports its performance. As can be seen, $OLS^{ALL}$ has the highest relative $R^2_{OOS}$ for the first three forecast horizons among all OLS-based models in Table 2. At the quarterly horizon, however, the relative $R^2_{OOS}$ remains negative at $-0.6\%$, which is worse than several individual $RV$-based models. The result might reflect the fact that, at the quarterly horizon, effective sample size drops significantly and thus we do not have enough data to estimate a dense OLS model with 118 features. Specifically, our forecasting models are designed to use as much data as possible by fitting daily updated $RV$'s on daily updated features for all forecast horizons. Because of overlapping data, however, the effective sample size of the data at the quarterly horizon is only about 1/63 of the sample size at the daily horizon. In such a case, we may need sparse or more regularized models. Another point worth noting is that $OLS^{ALL}$ outperforms $OLS^{RM}$ at the first three horizons, indicating the additional information contained in $IV$ measures.

### *4.2. Machine-Learning-Based Models*

Having established the initial evidence that increasing the number of features can improve forecast performance through a simple OLS fit, we now show that performance can be further improved by using learning algorithms other than OLS. Table 3 presents the $R^2_{OOS}$'s relative to HAR for the five learning algorithms discussed in Section 3, and for an ensemble model based on the five individual machine learning models (AVG).[16] Each model is trained using all 118 realized and implied variance features, so $OLS^{ALL}$ serves as a natural baseline. The second column of Table 3 lists the hyperparameters of each model with tuning parameters in bold, and in the last column we report the $R^2_{OOS}$'s relative to HAR. The most obvious pattern is that all machine learning models outperform HAR with positive relative $R^2_{OOS}$'s across the board. We then begin assessing the out-of-sample performance of each of the five machine learning models.

#### *4.2.1. Linear machine learning models*

First, we focus on the two linear learning algorithms LASSO and PCR. The row labeled "LASSO" and "PCR" in Table 3 presents their $R^2_{OOS}$'s relative to HAR. For LASSO, we validate its shrinkage parameter $\lambda$ from a set of 100 distinct values that covers a wide range of sparsity levels in the

[16] $R^2_{OOS}$'s relative to the long-run mean of $RV$'s for machine-learning-based models are presented in Table A.3 in the Appendix.

corresponding LASSO estimates of regression coefficients. For PCR, we validate the number of principal components as any integer between 1 and 20. The sparsity-encouraging $LASSO$ model has higher relative $R^2_{OOS}$'s than the unregularized $OLS^{ALL}$ across all forecast horizons, indicating the importance of sparsity in enhancing the out-of-sample performance. The dimension-reduction PCR approach underperforms LASSO at the daily, weekly, and monthly forecast horizons, but exhibits better performance at the quarterly forecast horizon with a relative $R^2_{OOS}$ of 7.8%. Why does PCR underperform at shorter horizons? Given that $IV$ features account for 86% of total features (102 out of 118), the first few principal components likely put more weights on $IV$ features and thus tend to better predict longer horizon $RV$'s due to the longer maturities of $IV$'s. Next, we turn our attention to the three nonlinear learning algorithms: RF, GBRT, and NN.

#### *4.2.2. Nonlinear machine learning models*

To train RF, we set the total number of trees to be 500 and use a subsample of 50% of the observations randomly drawn from each training sample (i.e., subsample = 0.5). At each node split, we randomly select 5 out of the 118 features (i.e., subfeature = $ln(118) = 5$). Subsample and subfeature can help decorrelate the trees to reduce overfitting. The maximum tree depth across all trees $L$ is a tuning parameter, which can take any integer value between 1 and 20. The relative $R^2_{OOS}$ of RF from Table 3 is at 3.2% for the daily forecast horizon and at 6.4% for the weekly forecast horizon, both of which are below the corresponding metrics of $OLS^{ALL}$. However, RF outperforms $OLS^{ALL}$ at the monthly and quarterly forecast horizons with relative $R^2_{OOS}$'s at 9.5% and 5.4%, respectively.

To train GBRT, we impose two early-stopping rules (whichever is met first): 1) when the MSE of the model does not decrease after 50 consecutive iterations, and 2) when the total number of trees reaches 20,000. Both the number of trees $B$ and the maximum tree depth are tuning parameters that we adaptively choose in the validation step, and the maximum tree depth can take any integer value between 1 and 5. For the remaining hyperparameters, we set the learning rate to be 0.001; to grow each tree, we randomly draw 50% of the observations from the training sample; at each node split, we randomly select 5 out of the 118 features (i.e., subfeature = $ln(118) = 5$). Overall, GBRT underperforms $OLS^{ALL}$ at the daily and weekly forecast horizons with relative $R^2_{OOS}$'s equal to 4.7% and 10.2%, but significantly outperforms $OLS^{ALL}$ at the monthly and quarterly horizons with

relative $R^2_{OOS}$ equal to 10.8% and 6.3%.

The performance of the two tree-based models improves as the forecasting horizon increases, which is likely driven by the strong predictive power of $IV$ features over longer forecast horizons. To see this, for both models, we only consider a random set of $ln(P)$ features at each tree split, and $IV$ features are more likely to be included in tree-based models than realized features given their dominance in the feature set. At shorter forecast horizons, these $IV$ features have limited predictive power because of their longer maturities between one and three months. At longer forecast horizons, however, the maturity of $IV$ features becomes more aligned with the forecast horizon, and $IV$ features included in the tree-based models become more powerful in predicting future $RV$'s.

To train NN, we consider two hidden layers with five and two neurons, respectively.[17] We choose the popular rectified linear unit (ReLU) as the activation function. In general, NN performs fairly well with relative $R^2_{OOS}$ equal to 10.5%, 16.7%, 14.3%, and 4.8% at the daily, weekly, monthly, and quarterly forecast horizons, respectively. NN delivers the best performance across all models at the first three horizons, but loses its dominance at the quarterly horizon. One potential explanation is that features interact less with each other or contribute to $RV$ more linearly at this horizon.[18]

#### *4.2.3. An ensemble model*

Comparing the out-of-sample performance of the five learning algorithms, we find that no single model strictly dominates the others. We then consider an ensemble model that combines volatility forecasts from different models (see, e.g., Timmermann, 2006, for an extensive survey of forecast combination.) We take a simple average of the five volatility forecast models and name the model as AVG. The motivation is that averaging forecasts from different models can improve the robustness of the model and reduce forecast variance. The out-of-sample performance of AVG shown at the bottom of Table 3 is indeed superior. This average model outperforms the first four individual machine learning models at each forecast horizon by a significant margin. Although the performance of AVG is comparable to or slightly weaker than that of NN at daily to monthly horizons, it significantly dominates NN at the quarterly horizon with an improvement in $R^2_{OOS}$ relative to HAR of more

[17] The results are robust to alternative NN structures as discussed in Section 5.1.

[18] To help generate insights into model complexity, Figure A.3 in the Appendix displays the chosen tuning parameters of LASSO, PCR, RF, and GBRT for each forecast horizon and validation period.

than 5%.[19] Overall, the relative $R^2_{OOS}$ of AVG ranges from 9.0% at the daily forecast horizon to up to 15.2% at the monthly forecast horizon, further highlighting the advantage of combining machine learning models in forecasting $RV$'s. In Table 4, we report the Diebold-Mariano (DM) $t$-statistics for pairwise comparisons of performance for a model in the row versus a model in the column; the magnitude of the DM statistics map to $p$-values in the same fashion as regression $t$-statistics. The simple average of all machine learning models AVG performs very well across all forecast horizons at the 1% or 5% significance level in most cases.

### *4.3. Performance Over Time*

To assess the relative performance of each model over time, we further divide the 2001–2019 testing sample into three subperiods (2001-2007, 2008-2009, and 2010-2019) and calculate the $R^2_{OOS}$'s relative to HAR for both OLS-based and ML-based models. In Table 5 we summarize the out-of-sample performance of all models over the three subperiods.

Panel A reports the results for the pre-crisis period between January 2001 and December 2007. Among OLS-based models, $OLS^{ALL}$ performs the best at daily, weekly, and monthly forecast horizons with relative $R^2_{OOS}$'s between 5.2% and 8.5%; at the quarterly forecast horizon, HARQ-F has the highest relative $R^2_{OOS}$ at 6.7%. Turning to the ML-based models, AVG exhibits superior performance across all forecast horizons, with relative $R^2_{OOS}$'s ranging from 6.6% to 20.5%.

Panel B shows the out-of-sample performance of all models between January 2008 and December 2009, the period covering the financial crisis and its aftermath. $OLS^{ALL}$ continues to outperform the remaining OLS-based models at the daily and weekly forecast horizons, whereas MIDAS dominates at the monthly horizon and HARQ-F beats other OLS-based models at the quarterly horizon. Among ML-based models, NN performs the best at the daily forecast horizon with a relative $R^2_{OOS}$ equal to 13.4%, and it performs on par with LASSO at the weekly horizon with a relative $R^2_{OOS}$ equal to 14.9%. LASSO dominates at the monthly horizon with a relative $R^2_{OOS}$ of 9.5%. At the longer quarterly horizon, PCR achieves the highest relative $R^2_{OOS}$ equal to 5.7%. The overall winning model remains, however, the ensemble model AVG. Although AVG cannot beat all of the stand-alone models at a given forecast horizon, it consistently delivers top performance across horizons.

[19] In Section 5.2, we use Elastic Net to train optimal weights for combining individual forecasts and find that the resulting combination forecast $AVG^{ENet}$ outperforms NN at daily, weekly, and quarterly horizons.

Panel C presents the relative $R^2_{OOS}$'s for each model during the post-crisis period between January 2010 and June 2019. Interestingly, the performance of $OLS^{ALL}$ during this period is quite impressive with relative $R^2_{OOS}$'s equal to 7.5%, 13.5%, 20.2%, and 15.1% at the daily, weekly, monthly, and quarterly horizons. In contrast, the relative $R^2_{OOS}$'s of the popular $RV$ forecasting models are all no greater than 6%. A natural question is, what explains the stellar performance of $OLS^{ALL}$? We conjecture that much of the gain comes from the better quality of the implied variance features in recent years. For example, the average daily dollar trading volume for stock options has increased steadily over the past two decades, implying that the overall option market is becoming more efficient in incorporating information about future price movements.[20] To mention more direct evidence, the relative $R^2_{OOS}$'s of $OLS^{IV}$ during the post-crisis period all become positive, in sharp contrast to the mostly negative values in the pre-crisis and crisis periods. This trajectory sheds additional light on the importance of including implied variance predictors in forecasting $RV$'s. Meanwhile, the ML-based models exhibit even more superior predictive power across forecast horizons than $OLS^{ALL}$ in the last decade. In particular, the ensemble model AVG is associated with relative $R^2_{OOS}$'s of 10.4%, 18.7%, 29.6%, and 27.5% at daily, weekly, monthly, and quarterly horizons, all dominating $OLS^{ALL}$ by significant margins. Taken together, the subsample results further highlight the importance of using machine learning techniques to exploit the rich information content in the giant feature set.

### *4.4. Individual Fitting vs. Panel Fitting*

In our main analyses, we purposely fit each model using panel data. Several recent studies rely on time-series models to extract return patterns.[21] For volatility prediction problems, however, using panel data instead of time-series data can increase estimation efficiency, given the well-known commonalities in the dynamic dependencies of volatilities and spillover effects across stocks.[22] To illustrate the benefits of using panel data, we compare the performance of time-series fitting vs.

[20] Figure 2 of Christoffersen et al. (2018) demonstrates a decline in the effective relative spreads for options throughout their sample period between January 2004 and December 2012, providing further evidence that the option market is becoming more efficient.

[21] See, e.g., Guijaro-Ordonez, Pelger, and Zanotti (2022), Jiang, Kelly, and Xiu (2022), and Murray, Xiao, and Xia (2022).

[22] Volatility spillover effects and commonalities in the dynamic dependencies are well documented in the traditional GARCH and stochastic volatility models, see Andersen, Bollerslev, Christoffersen, and Diebold (2006), Connor, Korajczyk, and Linton (2006), Taylor (2007), and the references therein. Recent work by Herskovic, Kelly, Lustig, and Van Nieuwerburgh (2016) and Bollerslev, Hood, Huss, and Pedersen (2018) further highlights the co-movement of stock volatilities over time.

panel fitting in Table 6. Specifically, we report the $R^2_{OOS}$'s relative to the long-run mean of $RV$ for six OLS-based volatility forecasting models fitted stock by stock (Panel A) or using panel data (Panel B).[23] Across all models and forecast horizons, the $R^2_{OOS}$'s based on panel data are consistently higher than those based on individual time series. The contrast is particularly remarkable for $OLS^{RM}$ with more predictors and for the longer quarterly forecast horizon when the effective sample size is smaller, consistent with the findings of Bollerslev, Hood, Huss, and Pedersen (2018) that the use of a panel-based estimation technique that explores risk similarity across stocks can enhance the efficiency of individual forecasts.[24]

To further visualize the difference between individual and panel fitting, Figure 1 presents a scatterplot of the average monthly realized and predicted variances from individual and panel $OLS^{RM}$ models for each of the stocks in our S&P 100 sample. The reported average realized variances for each of the stocks are simply calculated as the time-series mean of the realized variances. As the figure shows, the panel-fitting-based predictions are in general much closer to the 45-degree line than the individual-based predictions for $OLS^{RM}$ models at monthly forecasting horizon. Panel fitting also produces predictions less spread out than individual fitting, suggesting it can greatly reduce estimation variance by exploiting the commonality of volatility dynamics across stocks.

### *4.5. Model Implied Temporal Dependency*

The presence of both linear and nonlinear model specifications in our automated system complicates the interpretation of predictions, particularly when examining the temporal dependency structure utilized in forecasting volatility. However, whereas the actual structure is not directly observable, the dynamics of the model forecasts may be meaningfully inferred by linearly projecting the forecast to the space of past daily $RV$'s. Specifically, we regress our $RV$ forecasts from the AVG model on 63 lagged daily $RV$'s using the full out-of-sample evaluation period spanning from January 2001 to June 2019. Panel A of Figure 2 presents the implied weights, represented as regression coefficients,

[23] We compare the $R^2_{OOS}$'s relative to the long-run mean of $RV$ instead of these relative to the HAR model prediction given the performance of the latter may be affected by the fitting scheme, as shown in the first row of Table 6.

[24] In Section A.4.1 of the Appendix, we delve into further analyses of the factor structure in volatilities. Our findings reveal that the first few latent factors from PCA are unable to account for the majority of cross-sectional variations effectively. Moreover, we discover that relying on a common volatility factor for volatility prediction does not enhance the forecasting performance.

for the AVG model across forecast horizons. For comparison purposes, we also plot the weights assigned to lagged $RV$'s in the HAR, HExp, and MIDAS models for the monthly forecast horizon over the same period in Panel B.

As shown in Panel A, our system assigns greater weight to initial lags when the forecast horizon is shorter. The decay rates are also higher for shorter horizons, aligning with our conjecture that short-term signals hold greater influence in shorter forecast horizons. Yet the temporal dependencies do not change too dramatically from lag to lag, underscoring the overall interpretability of our automated system. Turning to Panel B, when predicting monthly $RV$, the implied weights are generally fairly close across models. Nonetheless, the HAR, HExp, and AVG models exhibit slightly "faster" decay than the MIDAS model, with a more rapid initial decay and less weight assigned to intermediate lags ranging from two days to two weeks. Among these three models, the AVG model is the "slowest" with less weight allocated to the initial lags and a smoother decay. Our findings suggest that, although seemingly similar, the dependency structure of the AVG model differs from existing models, potentially contributing to its superior out-of-sample performance.

### *4.6. Group Importance*

We rely on the group importance measure described in Section 3.5 to assess the importance of each group of features for forecasting future $RV$'s while simultaneously controlling for the rest of the feature groups. Specifically, we divide the 118 features into three groups. The first group "MIDAS & ExpRV" includes the $MIDAS$ feature, $ExpRV^1$, $ExpRV^5$, $ExpRV^{25}$, $ExpRV^{125}$, and $ExpGlRV$, all of which are smoothly weighted sums of lagged daily $RV$'s over longer periods. The second group "RV& RQ" includes $RV^d$, $RV^w$, $RV^m$, $RV^q$, $RVP^d$, $RVN^d$, $RV^d\sqrt{RQ^d}$, $RV^w\sqrt{RQ^w}$, $RV^m\sqrt{RQ^m}$, and $RV^q\sqrt{RQ^q}$, all of which are based on simple $RV$ and/or $RQ$ terms. And the third group "Implied Variance" includes 102 implied variance features. For each forecast horizon and each model, we estimate the reduction in $R^2$ by permutating all values of a feature group within each training sample, and then average the reductions in $R^2$ over all training samples to obtain a single group importance measure. As our group importance measures for each forecast horizon are normalized to sum to one, we can interpret the importance measure of each group as its relative contribution to the overall importance in percentage.

Panels A to F in Figure 3 display the group importance across various forecast horizons for

LASSO, PCR, RF, GBRT, NN, and AVG, respectively. This figure reveals several interesting findings. First, all three groups of features contribute significantly to the forecast across different horizons and learning algorithms. For example, each feature group contributes at least 10% in almost all settings with the only exception being "RV& RQ" for fitting LASSO at monthly and quarterly horizons. Second, the MIDAS and ExpRV terms tend to be more important for LASSO, jointly contributing around 70% to the overall prediction. This might be because the MIDAS and ExpRV features are already well-engineered through smoothing and denoising the raw data. Unlike the raw features such as $IV$'s, MIDAS and ExpRV terms can be viewed as competent encoders that represent the predictive structure in the data and thus are more likely to be directly picked up by linear models. Third, the implied variance features become more important over forecasting horizons. Since $IV$-based features all have maturities between one and three months, it may not be surprising that they can better predict longer-term $RV$'s. Fourth, $IV$ features are particularly important for tree-based models RF and GBRT. As discussed in Section 4.2.2, tree models only consider a random set of $ln(P)$ features at each tree split, and given the dominant presence of $IV$ features in the feature set, they are more likely to be included as predictors. As $IV$ features are more important in predicting longer horizon $RV$, the relatively poorer performance of RF and GBRT at daily and weekly horizons can be explained by the higher weights tree models put on $IV$ features and lower weights on well-engineered features such as MIDAS and ExpRV.

To further assess the relative importance of each feature group over time, we focus on AVG and report its group importance based on 118 features across forecast horizons for each training sample in our out-of-sample analyses. Our first training sample is from January 1996 to December 1999, and our last training sample is from January 1996 to December 2017. For each training sample and each feature group, we calculate group importance based on the reduction in $R^2$ from permutating all values of each feature within that group, and then normalize group importance per each training sample and each forecast horizon to sum to one. Figure 4 displays the group importance for each training sample. Overall, implied variance features grow increasingly important over time for all forecast horizons. At the daily (weekly) forecast horizon, the group importance of "IV" terms increases from 13.52% (14.56%) for the first training sample to 49.14% (55.92%) for the last training sample, and the trend is similar at longer monthly and quarterly forecast horizons. In the meanwhile, the importance of the other two feature groups shrinks significantly over time. For

example, at the monthly forecast horizon, the importance of "MIDAS & ExpRV" terms decreases from 60% by the end of year 1999 to 25.86% by the end of year 2017, and that of "RV& RQ" terms decreases from 23.5% to 15.39%. These observations are consistent with the stellar performance of the pure $IV$-based OLS model during the most recent period reported in Panel C of Table 5, and further highlight the crucial role implied variance features play in forecasting realized variances as the option market becomes more efficient.

The above method of group importance fits each model at the full model specification. One drawback is that it has estimation bias when using the estimates from the full model for subset selection. An alternative is best subset selection where we fit each subset of features separately and compare their OOS performance. Due to the limit of computing resources, we cannot perform the best subset selection for all subsets and horizons. Instead, we focus on all of the seven possible combinations of the three feature groups at the monthly horizon. Specifically, we refit each of the seven combinations of groups for each of the learning algorithms and each training-validation-testing sample and then compute their AVG performance.

Panel A Figure 5 shows the $R^2_{OOS}$ of these subsets. As can be seen, the full model with all three groups is the best model with the highest relative $R^2_{OOS}$ of 15.2%. Two-group models are better than one-group models. Based on these numbers, we then attribute the overall relative $R^2_{OOS}$ to each group using a modified SHAP measure of Lundberg and Lee (2017).[25] Panel B shows the decomposition of $R^2_{OOS}$ into the three feature groups. In particular, among the overall $R^2_{OOS}$ of 15.2%, "MIDAS & ExpRV", "RV& RQ", and "IV" contribute approximately 4.1%, 5.0% and 6.0%. All three groups contribute significantly to the OOS performance, which is generally consistent with the conclusion that all three groups are important based on the previous permutation-based group importance study.

[25]The SHAP measure computes the importance of a feature group as a weighted average of all the differences in $R^2_{OOS}$ when the model is trained with and without the feature group. Section A.4.3 in the Appendix provides details for our adaptation of the SHAP measure.

## 5. Robustness Studies

### *5.1. Alternative NN Structures*

Our baseline Neural Network (NN) model consists of two hidden layers with 5 and 2 neurons. To understand whether the performance of NN is sensitive to the choice of its structure, we consider three alternatives: a single-hidden-layer network with 2 neurons (NN1), a two hidden-layer network with 4 and 2 neurons (NN2), and a three-hidden-layer network with 8, 4, and 2 neurons (NN3). Other than the architecture, we keep the feature set, the activation function, and the training scheme the same as in Table 3 for these alternative NN models.[26] Panel A of Table 7 reports the out-of-sample performance of the alternative NN models along with the baseline NN model from Table 3 for easy comparison. Overall, the performance of these models is generally inline with the baseline model. For instance, the $R^2_{OOS}$'s at weekly forecasting horizon range from 14.9% for NN1 to 17.4% or NN2, comparable to the $R^2_{OOS}$ at 16.7% for the baseline. The results suggest that the superior out-of-sample performance of NN is robust to the choice of different structures.

### *5.2. Alternative Ensemble Methods*

Our ensemble model AVG is an equal-weighted average of forecasts from five individual machine learning algorithms. The benefit of such an approach is its simplicity, but one may wonder if the performance of the ensemble model may be further improved by using more sophisticated combination methods. We now consider four alternative ways to combine signals and present the results in Panel B of Table 7. The original ensemble model AVG is shown in the top row for easy comparison. These alternative combination methods include: 1) median of forecasts from five individual machine learning models (MED); 2) simple average of forecasts after removing the highest and lowest individual forecasts ($AVG^{Trim}$); 3) weighted average of forecasts from five individual models with weights equal to the inverse of the validation set MSE, where weights are normalized to sum up to one ($AVG^{ValidError}$); and 4) weighted average of forecasts from five individual models with weights tuned by Elastic Net ($AVG^{ENet}$). The first two ensemble methods are introduced to exclude extreme individual forecasts as inputs, while the third and fourth methods are designed

[26] We always use multiple random states when implementing stochastic optimization for estimation and derive predictions by averaging forecasts based on all tuned neural network models with ten starting points for more reliable estimates.

to optimize the weights on individual forecasts. To construct weights for $AVG^{ENet}$, we use a rolling-window training-validation-testing scheme for hyperparameter tuning. Specifically, we rolling fit Elastic Net (ENet) model with 1-year data, validate model hyperparameters with 3-month data, and predict the subsequent one-month $RV$ by combining five individual forecasts with weights from tuned ENet.[27] Therefore, we use the data between January 2001 and March 2002 as our first training and validation sets for $AVG^{ENet}$. To render the ensemble methods directly comparable with each other, we set the out-of-sample prediction period for all ensemble models to be between April 2002 and June 2019, so the different $R^2_{OOS}$'s for AVG in Table 7 from these in Table 3 are due to sample difference.

Overall, the first three ensemble methods MED, $AVG^{Trim}$, and $AVG^{ValidError}$ perform on par with AVG, with $R^2_{OOS}$'s similar to AVG across all forecasting horizons. For example, $AVG^{ValidError}$ produces the same $R^2_{OOS}$'s as AVG at daily and monthly horizons; its $R^2_{OOS}$ differs from AVG by only 0.2% at the weekly horizon and by 0.1% at the quarterly horizon. In contrast, $AVG^{ENet}$ generates $R^2_{OOS}$'s that are 1% to 3% higher than AVG at daily, weekly, and quarterly horizons, yet it produces an $R^2_{OOS}$ that is 1% lower than AVG at the monthly horizon. These numbers illuminate the advantage of Elastic Net in combining the signals, although such an advantage is not universal across horizons. The results are largely consistent with the extensive literature on model averaging in finance and economics. In particular, Timmermann (2006) highlights the effectiveness of using simple averages as opposed to more complex schemes of forecast combinations. This effectiveness can be attributed to the absence of estimation biases, particularly when the optimal weights are time-varying. Despite the fairly strong empirical performance of the simple average approach, we do not explicitly endorse this method. Instead, our aim is to establish a baseline that represents the lower limit of machine learning algorithms in volatility forecasting.

### *5.3. Firm Characteristics and Pure Noise*

Using the $RV$- and $IV$-based features, we have shown our volatility forecast system is able to achieve superior out-of-sample performance. We now consider two new feature sets: firm characteristics and pure noises. In the volatility forecasting literature, firm characteristics have not been widely

[27] We impose two simple restrictions on the ENet parameters to ensure that the resulting ensemble method is inline with the other ensemble methods: 1) the intercept is zero, and 2) the coefficients (weights) on five individual forecasts are non-negative.

documented as useful predictors.[28] On the other hand, it is reasonable to hypothesize that firm characteristics such as size might be indirectly (through interaction or nonlinearity) helpful in volatility forecasting. To examine the predictive power of firm characteristics, we consider the six representative features: $Size$, $BM$, $Mom$, $Ret^d$, $Ret^m$, and $ILLQ$. Following Kelly, Pruitt, and Su (2019) and Gu, Kelly, and Xiu (2020), we cross-sectionally rank each characteristic on each day and map these ranks into the [-1,1] interval. Then we use the relative ranks of these characteristics as additional features. The second new feature set is pure noise, with which we can test how well our system handles false positives. We generate six random noise terms that mimic the distributional properties of the volatility-based features. Section A.4.2 in the Appendix describes more details about these twelve additional features.

Panel C of Table 7 shows that the performance of the ensemble model AVG based on the original 118 features plus the 12 additional features. In comparison with the AVG results in Table 3, adding the new features shows minimal improvement over in the relative $R^2_{OOS}$'s at the first two horizons, identical relative $R^2_{OOS}$ at the monthly horizon, and slightly worse relative $R^2_{OOS}$ at the quarterly horizon.[29] Table A.4 in the Appendix reports the performance after adding the new features for each of the five learning algorithms. Overall, the augmented feature set generates very similar results to these using the original 118 features across different models.

We also calculate group importance of the two new feature sets. Figure A.4 in the Appendix displays the group importance plots based on the augmented 130 features for each individual ML model and the ensemble model AVG. The two new groups, "Firm Char" and "Noise," correspond to the six cross-sectionally ranked firm characteristics and six pure noise terms, respectively. There are several noteworthy patterns. First, the importance of the original three $RV$- and $IV$-based groups is largely aligned with what Figure 3 shows based on 118 features. Secondly, cross-sectionally ranked firm characteristics as a group contributes insignificantly to $RV$ prediction, with group importance ranging from 0.03% for LASSO at the quarterly horizon to 4.07% for RF at the same quarterly horizon. Lastly, the noise features contribute almost nothing to model prediction, indicating that our ML-based models and the associated group importance metrics effectively control for false

[28] Paye (2012) shows that volatility forecasts exploiting macroeconomic variables do not outperform a univariate benchmark out-of-sample much, and Rahimikia and Poon (2023) find that adding news sentiment variables only marginally improves the forecasting performance.

[29] The AVG results in Panels C and D of Table 7 are compared with the AVG results in Table 3 but not with the AVG in Panel B of Table 7, which are from results starting from a later year due to sample required for training $AVG^{ENet}$.

positives.

### *5.4. Human-engineered vs Raw features*

All of the 16 $RV$-based features in Section 2.3 are well-engineered by researchers based on lagged daily $RV$'s. Our volatility forecast system appears to be able to extract new information from them. It is then tempting to give the system a more challenging task: how would it perform if the input was just the raw daily $RV$'s and $IV$'s with no human-engineered features? To this end, we consider a new 165-variable feature set by replacing the 16 human-engineered $RV$-based features in the previous feature set of 118 variables with 63 lagged daily $RV$'s.

Panel E of Table 7 reports the out-of-sample performance of the ensemble model AVG using the new 165 features. The findings are quite intriguing. In comparison with the 118 feature results in Table 3, the relative $R^2_{OOS}$ based on the 165 raw features are quite close. At the daily and weekly horizons, the performance of the 165 raw features is only marginally worse than those of the 118 features (7.6% vs 9.1% for daily and 12.8% vs 14.3% for weekly). At the monthly and quarterly horizons, the performance of the raw features is close to or even better than those including the human-engineered features. Furthermore, when compared with the $OLS^{ALL}$ results in Table 2, the performance of these simple raw features under AVG is always better for all horizons, suggesting that the new system with just raw features can outperform the traditional OLS with well-engineered features. These results show that the proposed volatility forecast system is quite powerful even with the input as simple as the raw lagged RVs and IVs.

## 6. Predicting Realized Variances for S&P 500 Stocks

The previous sections demonstrate that our machine-learning-based automated system can improve volatility forecasting performance for the S&P 100 stocks. Is the learning system scalable to more stocks? In this section, we examine the out-of-sample performance of our system on a broader set of S&P 500 stocks. To speed up fitting nonlinear models, we transfer tuning parameters already learned from the original S&P 100 stock universe to the new S&P 500 universe. We find that tuning parameters learned from the original stock sample transfer well to the new sample and the resulting automated system consistently generates significant gains.

To this end, we consider 663 unique stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 500 index but are not members of the S&P 100 index between January 1996 and June 2019, and apply the same data filters as described in Section 2.1 to this stock sample. Because hyperparameter tuning for nonlinear models becomes more time-consuming as the sample grows, we directly transfer the tuning parameters for RF (i.e., maximum tree depth) and GBRT (i.e., # of trees and maximum tree depth) obtained from 173 S&P 100 stocks to 663 S&P 500 stocks, and retrain both models without validating these tuning parameters.[30] The idea is inspired by transfer learning, which is designed to explore the possibility that learned knowledge from one sample can be applied to a new sample.[31] The remaining ML-based as well as OLS-based models can be estimated efficiently and thus are completely recalibrated using the S&P 500 stock sample without hyperparameter transfer.

Table 8 summarizes the out-of-sample performance of all models for 663 S&P 500 stocks. Among OLS-based models, $OLS^{ALL}$ using all 118 predictors outperforms the remaining models at daily, weekly, and monthly forecast horizons with $R^2_{OOS}$'s relative to HAR between 4.9% and 8.6%. $OLS^{ALL}$ slightly underperforms HARQ-F at the quarterly forecast horizon but beats the rest of the OLS-based models across horizons, indicating that the 118 features identified earlier remain powerful volatility predictors for this broader universe. Note that the relative $R^2_{OOS}$'s of $OLS^{IV}$ become more negative between $-13.6\%$ and $-20.3\%$ for the S&P 500 stock sample in contrast to the relative $R^2_{OOS}$'s between $-2.1\%$ and $-9.8\%$ for the S&P 100 sample as reported in Table 2. This is likely because S&P 500 stocks tend to have fewer liquid option contracts than S&P 100 stocks as indicated by the fewer contracts in Table A.1 and thus the associated implied variance features are prone to measurement errors and biases. Yet, we find that implied variance features for S&P 500 stocks still contain information orthogonal to the realized features, as evident by the better performance of $OLS^{ALL}$ over that of $OLS^{RM}$ at each forecast horizon.

Turning to the ML-based models, LASSO outperforms $OLS^{ALL}$ across all horizons by small margins, while NN outperforms $OLS^{ALL}$ by wide margins with relative $R^2_{OOS}$'s ranging from 4.3% to 15.1%. The other three ML models PCR, RF, and GBRT produce out-of-sample performance comparable to $OLS^{ALL}$, indicating the success of hyperparameter transfer for the latter two

[30] All hyperparameters for NN are pre-specified and thus do not require hyperparameter tuning.

[31] Jiang, Kelly, and Xiu (2022) apply the image-based convolutional neural networks (CNNs) trained using daily data to lower-frequency and international data for return prediction problems. See Pan and Yang (2010) for a comprehensive survey on transfer learning.

models. For the ensemble model AVG, it consistently delivers higher relative $R^2_{OOS}$ than $OLS^{ALL}$ across forecast horizons, and the pairwise Diebold-Mariano $t$-statistics comparing the out-of-sample forecast performance between AVG and $OLS^{ALL}$ are all significant at the 1% level. In a nutshell, our automated system continues to perform well on this broader S&P 500 stock universe.

## 7. Economic Gains

We have demonstrated the statistical improvement in terms of relative $R^2_{OOS}$ achieved by our automated system for forecasting $RV$. A natural question is then to what extent can the increase in relative $R^2_{OOS}$'s translate into economic gains. In this section, we compare the return spreads of the variance risk premium (VRP) strategy based on different $RV$ forecasting models. Han and Zhou (2011) show that VRP, or the difference between expected variances under the risk-neutral measure and the physical measure, positively predicts cross-sectional stock returns.[32] We follow the literature and define VRP for stock $i$ on day $t$ based on $RV$ forecasting model $m$ as:

$$VRP^m_{i,t} = IV_{i,t} - E^m_t(RV_{i,t+21}), \tag{6}$$

where $IV_{i,t}$ is the annualized monthly implied variance from at-the-money call options with one-month maturity, and $E^m_t(RV_{i,t+21})$ is the annualized monthly expected realized variance from forecasting model $m$. Then we construct trading strategies based on VRP. Specifically, by the end of each day $t$ we sort stocks in our sample into decile portfolios based on their VRP on the same day, and compute value-weighted returns on each decile portfolio and a spread portfolio that buys stocks in the top decile with high VRP and sells stocks in the bottom decile with low VRP with a 21-day holding period.[33]

Panel A of Table 9 reports the average value-weighted monthly returns of the decile portfolios sorted by VRP based on different risk models for S&P 100 stocks, as well as the returns and alphas of the spread portfolios with Newey-West robust $t$-statistics with lag 20 in parentheses. We focus on the comparison among four models: 1) a perfect risk model, i.e., $E^m_t(RV_{i,t+21}) = RV_{i,t+21}$, 2) the simple

[32] Eraker and Wang (2015) propose a non-linear model to describe the VIX index and the variance risk premium. Eraker (2021) further studies a general equilibrium model based on long-run risk in an effort to explain the variance risk premium.

[33] Equal-weighted portfolios perform similarly to value-weighted portfolios.

OLS model using all features $OLS^{ALL}$, 3) the neural network model NN, and 4) the simple average ensemble model AVG. The row labeled "$RV_{t+21}$" reports the VRP portfolio performance based on perfect $RV$ forecasts. The monthly return spread monotonically increases from -2.08% to 1.80%, generating an annualized return spread of 45.6% (3.88% $\times$ 12) and a $t$-statistic of 11.92. This return pattern, although unrealistic, confirms the existence of cross-sectional VRP in our recent sample and sets the highest benchmark for other risk models. Turning to the performance of the VRP strategies based on forecasting models $OLS^{ALL}$, NN, and AVG, we continue to observe a generally increasing pattern in decile portfolio returns sorted by VRP, although not always monotonic. The return spread based on $OLS^{ALL}$ is 4.7% (0.39% $\times$ 12) per year, much lower than the annual return spread of 6.6% (0.55% $\times$ 12) based on NN and 8.5% (0.71% $\times$ 12) based on AVG. Thus, switching risk models from $OLS^{ALL}$ to the ensemble model AVG can increase the annual return of the VRP strategy by as much as 3.8%. We further find that the abnormal returns (alphas) on the VRP return spread are highly significant even after controlling for exposures to factors such as market (CAPM), or factors from the Fama-French 3- and 5-factor models (Fama and French, 1993, 2015).

We next examine whether the demonstrated economic values of our better $RV$ forecasts hold for the broader S&P 500 stock sample. Panel B of Table 9 repeats the VRP exercises and reports the average monthly returns of the decile portfolios sorted by VRP based on different risk models for the 836 S&P 500 stocks, including those in the main S&P 100 sample. Again, We focus on the comparison among four RV forecasting models. Consistent with our previous analyses on S&P 100 sample, decile portfolios constructed using S&P 500 stocks exhibit unrealistic yet most significant return spread with VRP from perfect $RV$ forecast "$RV_{t+21}$", with an annualized return spread of 51%(4.25% $\times$ 12) and a $t$-statistic of 12.82. Meanwhile, even though perfect forecasts are not accessible, the return spread is likely to be larger if model $m$ is good at predicting next month's $RV$. For VRP strategies based on forecasting models $OLS^{ALL}$, NN, and AVG, the annualized value-weighted return spreads for the S&P 500 sample range from 5.2% for $OLS^{ALL}$ to 8.6% for AVG and remain statistically significant after controlling for common risk factors. The much stronger VRP strategy profit from AVG again corroborates the superior predictive power of our automated forecasting system and illuminates the value of incorporating better $RV$ predictions into building stronger VRP strategies.

## 8. Conclusion

We propose an automated volatility forecasting system for 173 S&P 100 stocks using more than one hundred features and five machine learning algorithms. Our automated system is able to deliver superior out-of-sample volatility forecasting performance, compared to existing risk models that do not exploit the rich information embedded in big data. The performance of our system is particularly exceptional in the recent decade, indicating the importance of forecasting volatility via such a powerful system moving forward. In addition, our system remains robust to alternative specifications and is scalable to a broader S&P 500 stock universe via hyperparameter transfer learning. We further show that the improvement in out-of-sample prediction accuracy can translate into an enhanced annual return around 8.5% for a cross-sectional variance risk premium strategy.

On the methodological front, we propose feature engineering, model fitting, and evaluation methods specifically tailored to volatility forecasting problems. The pioneer work by Gu, Kelly, and Xiu (2020) provides important guidance on return prediction problems using machine learning. Our work not only offers detailed guidance on predicting volatility risk, but can also generate insights on forecasting other risk measures such as downside, crash, and tail risks.

Another promising research agenda is to translate visual data of volatility paths into volatility forecasts. Recently, Jiang, Kelly, and Xiu (2022) extract trading signals from price chart images and find that image-based price forecasts in general outperform traditional price trend signals. We expect that image-based learning would work well on volatility prediction for two reasons. First, compared with price trends, volatility trends are even more persistent. Second, several existing volatility forecasting models built on the decay functions of lagged volatilities (a smoothed version of volatility images) are proven to be powerful. Our work offers important implementation details on how to better map volatility images into predictions. All of these are important avenues for future research.

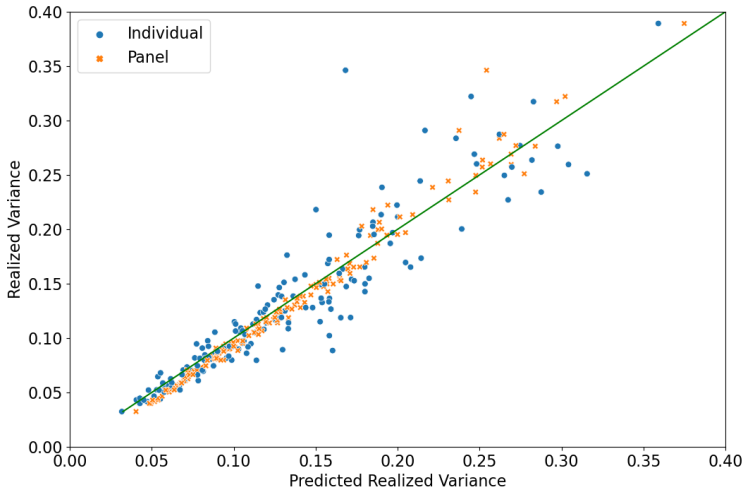



Fig. 1 Out-of-sample predictions: Individual v.s. panel fittings

The figure shows for each stock in our S&P 100 sample the average monthly realized variances (y-axis) against the average predicted monthly realized variances (x-axis) from $OLS^{RM}$ (i.e., simple OLS model with all 16 realized features as predictors) models between January 1996 and June 2019. The $OLS^{RM}$ models are either estimated on an individual stock-by-stock basis or estimated by panel regressions that restrict the coefficients to be the same across stocks.

*Panel A: Forecast horizons*

*Panel B: Risk models*

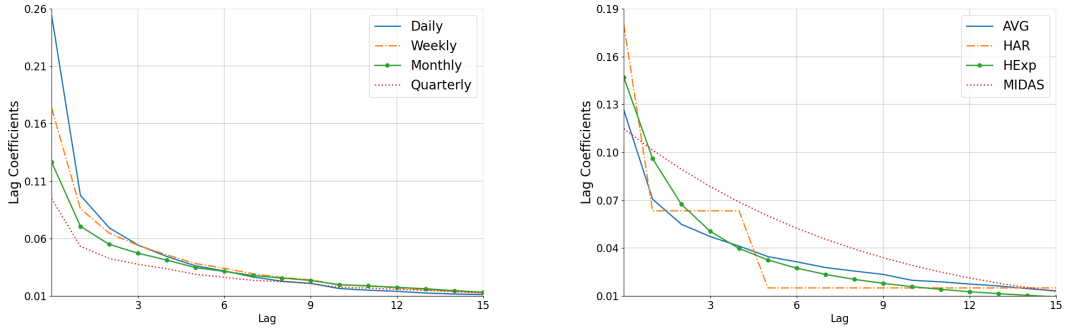
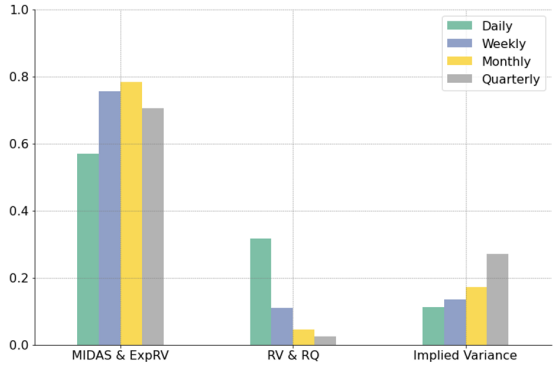



Fig. 2 Implied coefficients for different lags of $RV$'s

This figure displays the regression coefficients on lagged $RV$'s across different forecast horizons and predictive models. The regressions are conducted using the full out-of-sample evaluation period spanning from January 2001 to June 2019. The predictive models incorporated in the analysis include the AVG (a simple average of forecasts from the five individual machine learning models), HAR, HExp, and MIDAS models.

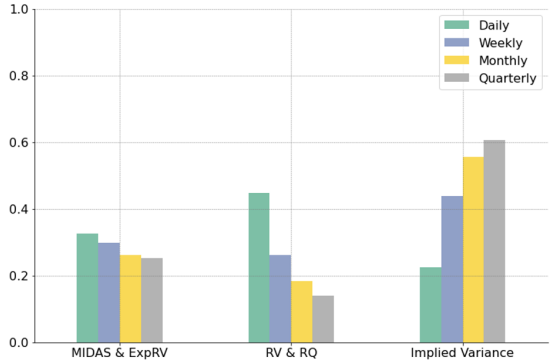
*Panel A: LASSO*

| | Daily | Weekly | Monthly | Quarterly |
|---|---|---|---|---|
| MIDAS & ExpRV | 56.93% | 75.60% | 78.35% | 70.49% |
| RV & RQ | 31.79% | 10.90% | 4.56% | 2.45% |
| Implied Variance | 11.28% | 13.50% | 17.09% | 27.06% |



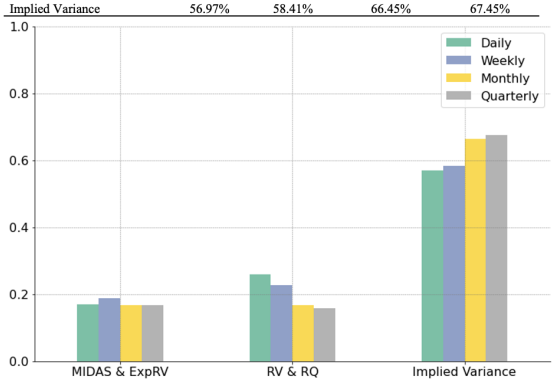

*Panel B: PCR*

| | Daily | Weekly | Monthly | Quarterly |
|---|---|---|---|---|
| MIDAS & ExpRV | 32.56% | 29.83% | 26.10% | 25.25% |
| RV & RQ | 44.84% | 26.23% | 18.41% | 14.04% |
| Implied Variance | 22.60% | 43.94% | 55.49% | 60.71% |



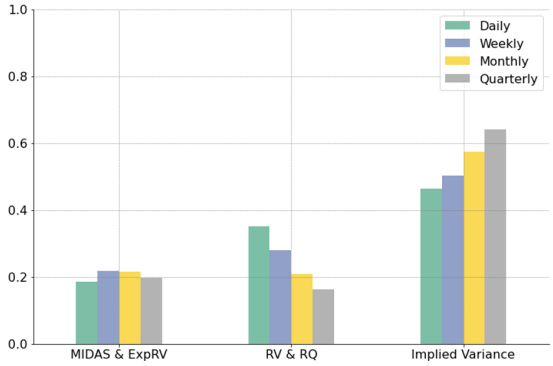

*Panel C: RF*

| | Daily | Weekly | Monthly | Quarterly |
|---|---|---|---|---|
| MIDAS & ExpRV | 17.04% | 18.85% | 16.71% | 16.69% |
| RV & RQ | 26.00% | 22.74% | 16.83% | 15.87% |
| Implied Variance | 56.97% | 58.41% | 66.45% | 67.45% |



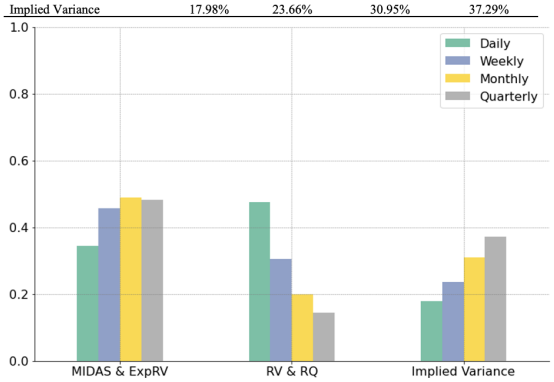

*Panel D: GBRT*

| | Daily | Weekly | Monthly | Quarterly |
|---|---|---|---|---|
| MIDAS & ExpRV | 18.62% | 21.70% | 21.62% | 19.66% |
| RV & RQ | 35.08% | 28.00% | 20.93% | 16.32% |
| Implied Variance | 46.30% | 50.31% | 57.45% | 64.03% |




*Panel E: NN*

| | Daily | Weekly | Monthly | Quarterly |
|---|---|---|---|---|
| MIDAS & ExpRV | 34.41% | 45.75% | 49.02% | 48.31% |
| RV & RQ | 47.61% | 30.59% | 20.03% | 14.40% |
| Implied Variance | 17.98% | 23.66% | 30.95% | 37.29% |




*Panel F: AVG*

| | Daily | Weekly | Monthly | Quarterly |
|---|---|---|---|---|
| MIDAS & ExpRV | 31.71% | 38.41% | 38.23% | 33.97% |
| RV & RQ | 36.25% | 23.57% | 15.62% | 13.03% |
| Implied Variance | 32.04% | 38.02% | 46.15% | 53.00% |

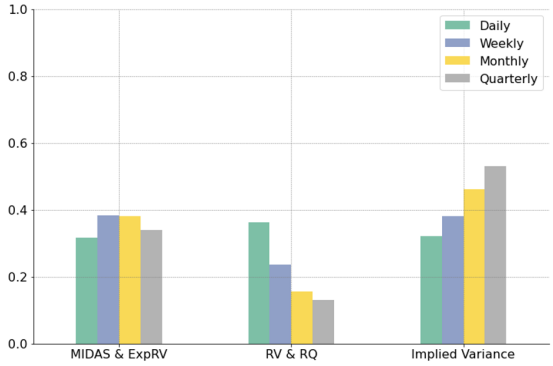



Fig. 3 Group importance based on 118 Features

This figure displays the group importance based on 118 features for LASSO, PCR, RF, GBRT, NN, and AVG across various forecast horizons. The first group "MIDAS & ExpRV" includes the $MIDAS$ term for the corresponding forecast horizon, $ExpRV^1$, $ExpRV^5$, $ExpRV^{25}$, $ExpRV^{125}$, and $ExpGlRV$. The second group "RV& RQ" includes $RV^d$, $RV^w$, $RV^m$, $RV^q$, $RVP^d$, $RVN^d$, $RV^d\sqrt{RQ^d}$, $RV^w\sqrt{RQ^w}$, $RV^m\sqrt{RQ^m}$, and $RV^q\sqrt{RQ^q}$. The third group "Implied Variance" includes $CIV^{jm,\delta}$ and $PIV^{jm,-\delta}$, where $j = 1, 2, 3$, and $\delta = 0.1, 0.15, ..., 0.9$.

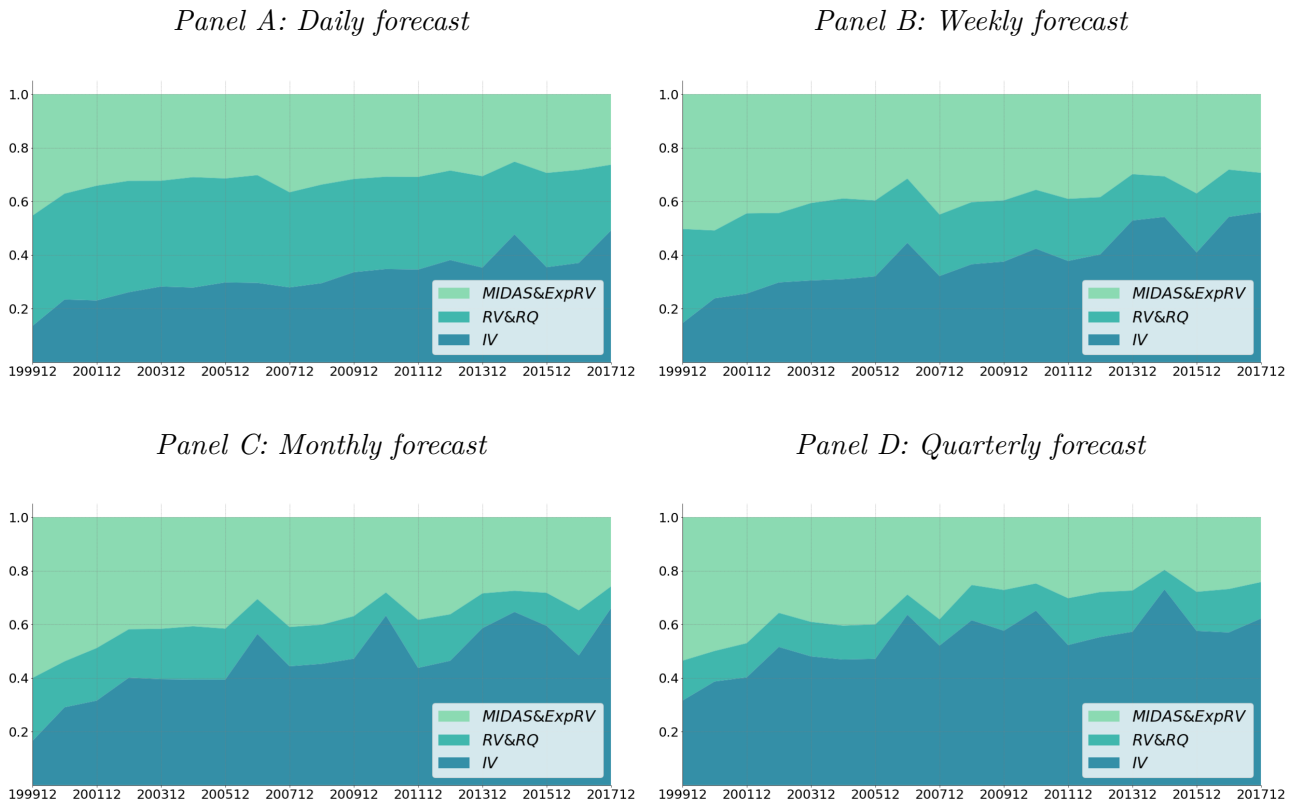



Fig. 4 Group importance for AVG over time

This figure displays the group importance based on 118 features for AVG across forecast horizons for each training sample in our out-of-sample analyses. Our first training sample is from January 1996 to December 1999, and our last training sample is from January 1996 to December 2017. The first group "MIDAS & ExpRV" includes the $MIDAS$ term for the corresponding forecast horizon, $ExpRV^1$, $ExpRV^5$, $ExpRV^{25}$, $ExpRV^{125}$, and $ExpGlRV$. The second group "RV& RQ" includes $RV^d$, $RV^w$, $RV^m$, $RV^q$, $RVP^d$, $RVN^d$, $RV^d\sqrt{RQ^d}$, $RV^w\sqrt{RQ^w}$, $RV^m\sqrt{RQ^m}$, and $RV^q\sqrt{RQ^q}$. The third group "Implied Variance" includes $CIV^{jm,\delta}$ and $PIV^{jm,-\delta}$, where $j = 1, 2, 3$, and $\delta = 0.1, 0.15, ..., 0.9$.

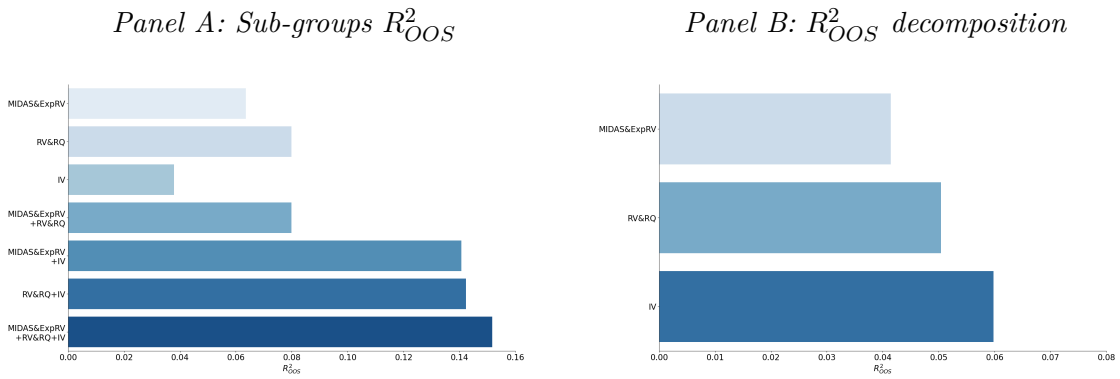



Fig. 5 Best subset selection with model refit

This figure displays the results for the sub-group analysis and decomposition of out-of-sample $R^2$ relative to HAR model for the ensemble model AVG at monthly forecast horizon. Panel A shows the out-of-sample $R^2$ relative to HAR model for AVG model based on subsets of feature groups. Panel B displays the decomposition of the total out-of-sample $R^2$ into group importance based on Eq. (A.18).

## Table 1 Feature correlation

This table reports the correlations of all realized features and selective implied variance features with absolute delta equal to 0.5. The sample consists of 173 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 100 index between January 1996 and June 2019 with share code 10 or 11, prices between \$1 and \$1000, daily number of trades greater than or equal to 100, and at least five years of data on all features and response variables. Superscripts $d$, $w$, $m$, and $q$ are abbreviations of daily, weekly, monthly, and quarterly construction intervals or forecast horizons. $MIDAS^k$ $(k = d, w, m, q)$ denotes the smoothly weighted moving average of 50 lagged realized variances using validated polynomials in forecasting realized variance at horizon $k$. $RV^k$ $(k = d, w, m, q)$ is the daily, weekly, monthly or quarterly realized variance. $RVP^d$ and $RVN^d$ are the daily realized positive and negative semivariances, respectively. $RV^k\sqrt{RQ^k}$ $(k = d, w, m, q)$ is the product of the realized variance and the square root of the realized quarticity with the same construction interval $k$. $ExpRV^i$ $(i = 1, 5, 25, 125)$ is the exponentially weighted moving average of the past 500-day realized variances using the corresponding center-of-mass $i$. $ExpGlRV$ is the exponentially weighted moving average of the global risk factor with a 5-day center-of-mass. $CIV^{jm,0.5}$ and $PIV^{jm,-0.5}$ are implied variances from call and put options with absolute delta equal to 0.5 and maturity equal to $j$ months $(j = 1, 2, 3)$.

| Variable | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) | (10) | (11) | (12) | (13) | (14) | (15) | (16) | (17) | (18) | (19) | (20) | (21) | (22) | (23) | (24) | (25) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (1) $MIDAS^d$ | 1.00 | | | | | | | | | | | | | | | | | | | | | | | | |
| (2) $MIDAS^w$ | 0.99 | 1.00 | | | | | | | | | | | | | | | | | | | | | | | |
| (3) $MIDAS^m$ | 0.98 | 1.00 | 1.00 | | | | | | | | | | | | | | | | | | | | | | |
| (4) $MIDAS^q$ | 0.96 | 0.99 | 1.00 | 1.00 | | | | | | | | | | | | | | | | | | | | | |
| (5) $RV^d$ | 0.86 | 0.82 | 0.80 | 0.77 | 1.00 | | | | | | | | | | | | | | | | | | | | |
| (6) $RV^w$ | 0.97 | 0.95 | 0.94 | 0.91 | 0.81 | 1.00 | | | | | | | | | | | | | | | | | | | |
| (7) $RV^m$ | 0.92 | 0.95 | 0.96 | 0.98 | 0.73 | 0.88 | 1.00 | | | | | | | | | | | | | | | | | | |
| (8) $RV^q$ | 0.82 | 0.86 | 0.88 | 0.90 | 0.65 | 0.77 | 0.89 | 1.00 | | | | | | | | | | | | | | | | | |
| (9) $RVP^d$ | 0.80 | 0.77 | 0.75 | 0.73 | 0.90 | 0.76 | 0.68 | 0.62 | 1.00 | | | | | | | | | | | | | | | | |
| (10) $RVN^d$ | 0.78 | 0.75 | 0.73 | 0.71 | 0.90 | 0.74 | 0.67 | 0.61 | 0.65 | 1.00 | | | | | | | | | | | | | | | |
| (11) $RV^d\sqrt{RQ^d}$ | 0.49 | 0.44 | 0.42 | 0.39 | 0.72 | 0.46 | 0.37 | 0.30 | 0.64 | 0.61 | 1.00 | | | | | | | | | | | | | | |
| (12) $RV^w\sqrt{RQ^w}$ | 0.70 | 0.67 | 0.65 | 0.62 | 0.59 | 0.78 | 0.61 | 0.48 | 0.54 | 0.52 | 0.54 | 1.00 | | | | | | | | | | | | | |
| (13) $RV^m\sqrt{RQ^m}$ | 0.67 | 0.70 | 0.71 | 0.71 | 0.53 | 0.66 | 0.79 | 0.63 | 0.49 | 0.47 | 0.39 | 0.71 | 1.00 | | | | | | | | | | | | |
| (14) $RV^q\sqrt{RQ^q}$ | 0.59 | 0.61 | 0.63 | 0.65 | 0.46 | 0.56 | 0.68 | 0.80 | 0.43 | 0.42 | 0.29 | 0.50 | 0.71 | 1.00 | | | | | | | | | | | |
| (15) $ExpRV^1$ | 0.96 | 0.93 | 0.91 | 0.88 | 0.93 | 0.95 | 0.84 | 0.74 | 0.85 | 0.84 | 0.59 | 0.72 | 0.63 | 0.53 | 1.00 | | | | | | | | | | |
| (16) $ExpRV^5$ | 0.98 | 0.98 | 0.98 | 0.97 | 0.82 | 0.96 | 0.95 | 0.84 | 0.76 | 0.75 | 0.45 | 0.72 | 0.74 | 0.62 | 0.94 | 1.00 | | | | | | | | | |
| (17) $ExpRV^{25}$ | 0.90 | 0.93 | 0.94 | 0.96 | 0.71 | 0.85 | 0.96 | 0.97 | 0.67 | 0.66 | 0.35 | 0.57 | 0.73 | 0.76 | 0.82 | 0.92 | 1.00 | | | | | | | | |
| (18) $ExpRV^{125}$ | 0.76 | 0.79 | 0.80 | 0.83 | 0.60 | 0.71 | 0.80 | 0.90 | 0.57 | 0.56 | 0.26 | 0.40 | 0.50 | 0.62 | 0.68 | 0.76 | 0.89 | 1.00 | | | | | | | |
| (19) $ExpGlRV$ | 0.62 | 0.63 | 0.64 | 0.64 | 0.50 | 0.58 | 0.60 | 0.57 | 0.48 | 0.47 | 0.20 | 0.28 | 0.30 | 0.29 | 0.56 | 0.60 | 0.60 | 0.59 | 1.00 | | | | | | |
| (20) $CIV^{1m,0.5}$ | 0.83 | 0.85 | 0.85 | 0.86 | 0.69 | 0.79 | 0.84 | 0.82 | 0.63 | 0.65 | 0.32 | 0.50 | 0.58 | 0.58 | 0.77 | 0.84 | 0.85 | 0.78 | 0.59 | 1.00 | | | | | |
| (21) $CIV^{2m,0.5}$ | 0.82 | 0.84 | 0.85 | 0.86 | 0.67 | 0.78 | 0.83 | 0.83 | 0.62 | 0.63 | 0.31 | 0.49 | 0.58 | 0.58 | 0.76 | 0.83 | 0.86 | 0.79 | 0.59 | 0.98 | 1.00 | | | | |
| (22) $CIV^{3m,0.5}$ | 0.82 | 0.84 | 0.85 | 0.86 | 0.67 | 0.77 | 0.84 | 0.84 | 0.62 | 0.62 | 0.31 | 0.48 | 0.58 | 0.59 | 0.75 | 0.82 | 0.87 | 0.81 | 0.60 | 0.97 | 0.99 | 1.00 | | | |
| (23) $PIV^{1m,-0.5}$ | 0.78 | 0.80 | 0.80 | 0.81 | 0.64 | 0.75 | 0.80 | 0.79 | 0.59 | 0.60 | 0.32 | 0.53 | 0.63 | 0.63 | 0.73 | 0.80 | 0.82 | 0.73 | 0.52 | 0.90 | 0.89 | 0.88 | 1.00 | | |
| (24) $PIV^{2m,-0.5}$ | 0.77 | 0.78 | 0.79 | 0.80 | 0.62 | 0.73 | 0.79 | 0.79 | 0.58 | 0.58 | 0.31 | 0.52 | 0.62 | 0.64 | 0.71 | 0.78 | 0.82 | 0.74 | 0.51 | 0.88 | 0.88 | 0.88 | 0.99 | 1.00 | |
| (25) $PIV^{3m,-0.5}$ | 0.75 | 0.77 | 0.78 | 0.79 | 0.61 | 0.72 | 0.78 | 0.79 | 0.57 | 0.56 | 0.31 | 0.50 | 0.61 | 0.64 | 0.69 | 0.77 | 0.81 | 0.74 | 0.50 | 0.85 | 0.86 | 0.87 | 0.98 | 0.99 | 1.00 |

Table 2 Out-of-sample prediction relative to HAR: OLS-based models

This table reports the out-of-sample $R^2$ relative to the HAR model for OLS-based volatility forecasting models across different forecast horizons. The sample consists of 173 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 100 index between January 1996 and June 2019 with share code 10 or 11, prices between \$1 and \$1000, daily number of trades greater than or equal to 100, and at least five years of data on all features and response variables. The full out-of-sample evaluation period is from January 2001 to June 2019. Superscripts $d$, $w$, $m$, and $q$ are abbreviations of daily, weekly, monthly, and quarterly construction intervals or forecast horizons. $MIDAS$ denotes the smoothly weighted moving average of 50 lagged realized variances using validated polynomials for the corresponding forecast horizon. $RV^k$ $(k = d, w, m, q)$ is the daily, weekly, monthly or quarterly realized variance. $RVP^d$ and $RVN^d$ are the daily realized positive and negative semivariances, respectively. $RV^k\sqrt{RQ^k}$ $(k = d, w, m, q)$ is the product of the realized variance and the square root of the realized quarticity with the same construction interval $k$. $ExpRV^i$ $(i = 1, 5, 25, 125)$ is the exponentially weighted moving average of the past 500-day realized variances using the corresponding center-of-mass $i$. $ExpGlRV$ is the exponentially weighted moving average of the global risk factor with a 5-day center-of-mass. $CIV^{jm,\delta}$ and $PIV^{jm,-\delta}$ are implied variances from call and put options with absolute $\delta = 0.1, 0.15, ..., 0.9$ and maturity equal to $j$ months $(j = 1, 2, 3)$. Our OLS-based models include MIDAS, SHAR, HARQ-F, HExpGl, $OLS^{RM}$ (i.e., simple OLS model with all 16 realized features as predictors), $OLS^{IV}$ (i.e., simple OLS model with all 102 implied variance features as predictors), and $OLS^{ALL}$ (i.e., simple OLS model with all 118 realized and implied variance features as joint predictors). $R^2_{OOS}$ for each model is calculated relative to the prediction from HAR using the entire panel of stocks according to Eq. (5).

| Model | Features | Daily | Weekly | Monthly | Quarterly |
|---|---|---|---|---|---|
| | | $R^2_{OOS}$ relative to HAR | | | |
| MIDAS | $MIDAS$ term for the corresponding forecast horizon | 1.1% | 3.8% | 4.4% | 1.5% |
| SHAR | $RVP^d$, $RVN^d$, $RV^w$, $RV^m$, $RV^q$ | 1.5% | 1.6% | 1.3% | 0.6% |
| HARQ-F | $RV^d$, $RV^w$, $RV^m$, $RV^q$, $RV^d\sqrt{RQ^d}$, $RV^w\sqrt{RQ^w}$, $RV^m\sqrt{RQ^m}$, $RV^q\sqrt{RQ^q}$ | 2.1% | 2.8% | 3.4% | 4.8% |
| HExpGl | $ExpRV^1$, $ExpRV^5$, $ExpRV^{25}$, $ExpRV^{125}$, $ExpGlRV$ | 0.1% | 2.6% | 2.2% | -1.4% |
| $OLS^{RM}$ | $MIDAS$ term for the corresponding forecast horizon, $RV^d$, $RV^w$, $RV^m$, $RV^q$, $RVP^d$, $RVN^d$, $RV^d\sqrt{RQ^d}$, $RV^w\sqrt{RQ^w}$, $RV^m\sqrt{RQ^m}$, $RV^q\sqrt{RQ^q}$, $ExpRV^1$, $ExpRV^5$, $ExpRV^{25}$, $ExpRV^{125}$, $ExpGlRV$ (# of features = 16) | 4.9% | 6.5% | 5.4% | 1.9% |
| $OLS^{IV}$ | $CIV^{jm,\delta}$ and $PIV^{jm,-\delta}$, $j = 1, 2, 3$, $\delta = 0.1, 0.15, ..., 0.9$ (# of features = 102) | -9.8% | -7.4% | -2.8% | -2.1% |
| $OLS^{ALL}$ | All 118 Features (16 realized features + 102 $IV$ features) | 7.6% | 11.6% | 7.3% | -0.6% |

Table 3 Out-of-sample predictions relative to HAR: Machine-learning-based models

This table reports the out-of-sample $R^2$ relative to the HAR model for machine-learning-based volatility forecasting models across different forecast horizons. The sample consists of 173 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 100 index between January 1996 and June 2019 with share code 10 or 11, prices between \$1 and \$1000, daily number of trades greater than or equal to 100, and at least five years of data on all features and response variables. The full out-of-sample evaluation period is from January 2001 to June 2019. The features of each model consist of all 118 predictors detailed in Table 2. Our machine-learning-based models include LASSO, Principal Component Regression (PCR), Random Forest (RF), Gradient Boosted Regression Trees (GBRT), Neural Network (NN), and a simple average of forecasts from the five individual machine learning models (AVG). Tuning parameters for each model are reported in **bold**. $R^2_{OOS}$ for each model is calculated relative to the prediction from HAR using the entire panel of stocks according to Eq. (5).

| Model | Hyperparameter (Tuning parameter in **bold**) | Daily | Weekly | Monthly | Quarterly |
|---|---|---|---|---|---|
| | | $R^2_{OOS}$ relative to HAR | | | |
| LASSO | **# of shrinkage parameters ($\lambda$): 100**<br>$\lambda_{min}/\lambda_{max}$: 0.001 | 8.0% | 12.1% | 11.3% | 2.6% |
| PCR | **# of components: 1, 2, ..., 20** | 5.5% | 4.8% | 8.1% | 7.8% |
| RF | **Maximum tree depth ($L$): 1, 2, ..., 20**<br># of trees: 500<br>Subsample: 0.5<br>Subfeature: ln(# of features) | 3.2% | 6.4% | 9.5% | 5.4% |
| GBRT | **# of trees ($B$)**<br>**Maximum tree depth ($L$): 1, 2, ..., 5**<br>Learning rate: 0.001<br>Subsample: 0.5<br>Subfeature: ln(# of features)<br>Early-stopping rules (whichever met first):<br>1) No reduction in MSE after 50 iterations<br>2) Max # of trees hits 20,000 | 4.7% | 10.2% | 10.8% | 6.3% |
| NN | # of hidden layer: 2<br># of neurons: (5, 2)<br>Activation function: ReLU | 10.5% | 16.7% | 14.3% | 4.8% |
| AVG | | 9.0% | 14.3% | 15.2% | 10.0% |

Table 4 Forecast comparison using Diebold-Mariano tests

This table reports pairwise Diebold-Mariano $t$-statistics comparing the out-of-sample forecast performance among seven models across different forecast horizons. The sample consists of 173 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 100 index between January 1996 and June 2019 with share code 10 or 11, prices between \$1 and \$1000, daily number of trades greater than or equal to 100, and at least five years of data on all features and response variables. The full out-of-sample evaluation period is from January 2001 to June 2019. The features of each model consist of all 118 predictors detailed in Table 2. Our models include a simple OLS model using all features ($OLS^{ALL}$), LASSO, Principal Component Regression (PCR), Random Forest (RF), Gradient Boosted Regression Trees (GBRT), Neural Network (NN), and a simple average of forecasts from the five individual machine learning models (AVG). Positive numbers indicate that the model denoted by the label to the left of a given row outperforms the model denoted by the label above the corresponding column. *, **, and *** indicate significance at the 10%, 5%, and 1% levels, respectively.

*Panel A: Daily forecast*

| | $OLS^{ALL}$ | LASSO | PCR | RF | GBRT | NN |
|---|---|---|---|---|---|---|
| LASSO | 3.30*** | | | | | |
| PCR | -7.02*** | -8.87*** | | | | |
| RF | -4.62*** | -5.06*** | -2.32** | | | |
| GBRT | -4.93*** | -5.88*** | -1.20 | 2.27** | | |
| NN | 10.67*** | 9.95*** | 13.09*** | 8.68*** | 11.97*** | |
| AVG | 4.75*** | 4.07*** | 10.16*** | 8.17*** | 12.28*** | -6.86*** |

*Panel B: Weekly forecast*

| | $OLS^{ALL}$ | LASSO | PCR | RF | GBRT | NN |
|---|---|---|---|---|---|---|
| LASSO | 1.05 | | | | | |
| PCR | -3.22*** | -3.36*** | | | | |
| RF | -2.22** | -2.42** | 1.15 | | | |
| GBRT | -0.90 | -1.48 | 3.07*** | 2.38** | | |
| NN | 7.36*** | 6.13*** | 6.10*** | 5.25*** | 5.88*** | |
| AVG | 2.64*** | 2.23** | 6.27*** | 5.16*** | 5.45*** | -3.69*** |

*Panel C: Monthly forecast*

| | $OLS^{ALL}$ | LASSO | PCR | RF | GBRT | NN |
|---|---|---|---|---|---|---|
| LASSO | 1.62 | | | | | |
| PCR | 0.19 | -1.12 | | | | |
| RF | 0.64 | -0.53 | 0.48 | | | |
| GBRT | 1.12 | -0.18 | 1.13 | 0.94 | | |
| NN | 3.25*** | 1.51 | 2.31** | 1.99** | 2.03** | |
| AVG | 2.63*** | 2.07** | 3.32*** | 2.80*** | 4.29*** | 0.75 |

*Panel D: Quarterly forecast*

| | $OLS^{ALL}$ | LASSO | PCR | RF | GBRT | NN |
|---|---|---|---|---|---|---|
| LASSO | 1.38 | | | | | |
| PCR | 2.71*** | 1.66* | | | | |
| RF | 1.92* | 0.89 | -0.68 | | | |
| GBRT | 1.95* | 1.30 | -0.50 | 0.46 | | |
| NN | 1.72* | 0.89 | -1.02 | -0.24 | -0.74 | |
| AVG | 3.09*** | 3.12*** | 0.83 | 1.78* | 2.71*** | 2.78*** |

Table 5 Out-of-sample prediction relative to HAR: Subsample analysis

This table reports the out-of-sample $R^2$ relative to the HAR model for OLS-based and machine-learning-based volatility forecasting models across different forecast horizons over three subsample periods. The sample consists of 173 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 100 index between January 1996 and June 2019 with share code 10 or 11, prices between \$1 and \$1000, daily number of trades greater than or equal to 100, and at least five years of data on all features and response variables. The full out-of-sample evaluation period is from January 2001 to June 2019. The features of each model consist of all 118 predictors detailed in Table 2. Our OLS-based models include MIDAS, SHAR, HARQ-F, HExpGl, $OLS^{RM}$ (i.e., a simple OLS model with all 16 realized features as predictors), $OLS^{IV}$ (i.e., a simple OLS model with all 102 implied variance features as predictors), and $OLS^{ALL}$ (i.e., a simple OLS model with all 118 realized and implied variance features as joint predictors). Our machine-learning-based models include LASSO, Principal Component Regression (PCR), Random Forest (RF), Gradient Boosted Regression Trees (GBRT), Neural Network (NN), and a simple average of forecasts from the five individual machine learning models (AVG). $R^2_{OOS}$ for each model is calculated relative to the prediction from HAR using the panel of stocks included in each subsample period according to Eq. (5). Panels A, B and C report $R^2_{OOS}$ relative to HAR for the pre-crisis (2001-2007), crisis (2008-2009), and post-crisis (2010-2019) periods, respectively.

| | | *Panel A: Pre-crisis (2001-2007)* | | | | *Panel B: Crisis (2008-2009)* | | | | *Panel C: Post-crisis (2010-2019)* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Daily | Weekly | Monthly | Quarterly | Daily | Weekly | Monthly | Quarterly | Daily | Weekly | Monthly | Quarterly |
| | | $R^2_{OOS}$ relative to HAR | | | | | | | | | | | |
| OLS | MIDAS | -0.4% | 1.0% | -2.4% | -0.9% | 3.9% | 7.1% | 8.3% | 2.1% | 0.4% | 3.1% | 5.1% | 4.3% |
| | SHAR | 1.1% | 1.3% | 1.1% | 0.6% | 2.1% | 2.1% | 1.5% | 0.6% | 1.5% | 1.2% | 1.1% | 0.8% |
| | HARQ-F | 1.9% | 3.0% | 3.9% | 6.7% | 3.4% | 3.6% | 3.2% | 4.0% | 1.1% | 1.4% | 3.0% | 5.3% |
| | HExpGl | 0.2% | 2.5% | 3.5% | 3.1% | -0.3% | 3.6% | 1.3% | -4.2% | 0.3% | 1.1% | 2.6% | 6.0% |
| | $OLS^{RM}$ | 4.0% | 5.8% | 4.3% | 2.3% | 6.6% | 7.2% | 4.8% | 0.6% | 4.4% | 6.3% | 10.2% | 10.3% |
| | $OLS^{IV}$ | -12.5% | -13.0% | -1.5% | 2.9% | -15.4% | -11.9% | -8.4% | -5.6% | 0.4% | 8.4% | 15.5% | 9.1% |
| | $OLS^{ALL}$ | 5.2% | 8.5% | 5.6% | -1.6% | 11.2% | 13.5% | 4.8% | -2.5% | 7.5% | 13.5% | 20.2% | 15.1% |
| ML | LASSO | 5.7% | 8.9% | 9.3% | 4.4% | 11.8% | 14.9% | 9.5% | -1.0% | 7.4% | 12.8% | 22.3% | 23.0% |
| | PCR | 2.6% | 7.0% | 8.7% | 8.1% | 10.1% | -2.2% | 4.5% | 5.7% | 5.3% | 12.1% | 20.0% | 21.9% |
| | RF | 0.6% | 1.9% | 9.9% | 7.9% | 0.0% | 2.0% | 4.0% | 1.6% | 10.8% | 20.2% | 29.2% | 25.3% |
| | GBRT | -0.5% | 3.6% | 9.2% | 10.9% | 7.6% | 11.7% | 7.0% | 1.9% | 9.8% | 18.1% | 28.4% | 24.3% |
| | NN | 8.1% | 16.6% | 20.2% | 16.1% | 13.4% | 14.9% | 6.6% | -2.2% | 11.1% | 19.6% | 30.1% | 23.4% |
| | AVG | 6.6% | 13.7% | 19.5% | 20.5% | 11.3% | 12.1% | 8.8% | 3.4% | 10.4% | 18.7% | 29.6% | 27.5% |

Table 6 Out-of-sample prediction relative to long-run mean: Individual v.s. panel fittings

This table reports the out-of-sample $R^2$ relative to the historical mean of realized volatilities for six OLS-based volatility forecasting models across different forecast horizons. The sample consists of 173 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 100 index between January 1996 and June 2019 with share code 10 or 11, prices between \$1 and \$1000, daily number of trades greater than or equal to 100, and at least five years of data on all features and response variables. The full out-of-sample evaluation period is from January 2001 to June 2019. Our OLS-based models include HAR, MIDAS, SHAR, HARQ-F, HExpGl, and $OLS^{RM}$ (i.e., simple OLS model with all 16 realized features as predictors). Panel A reports the out-of-sample $R^2$'s for OLS models estimated on an individual stock-by-stock basis. Panel B reports the out-of-sample $R^2$'s for OLS models estimated by panel regressions that restrict the coefficients to be the same across stocks. $R^2_{OOS}$ for each model at each forecast horizon is calculated relative to the long-run mean of $RV$ using the entire panel of stocks according to Eq. (5).

| Model | *Panel A: Individual fitting* | | | | *Panel B: Panel fitting* | | | |
|---|---|---|---|---|---|---|---|---|
| | Daily | Weekly | Monthly | Quarterly | Daily | Weekly | Monthly | Quarterly |
| | $R^2_{OOS}$ relative to long-run mean | | | | $R^2_{OOS}$ relative to long-run mean | | | |
| HAR | 56.63% | 64.04% | 61.18% | 51.64% | 57.8% | 69.4% | 70.0% | 63.6% |
| MIDAS | 57.27% | 65.87% | 63.32% | 53.80% | 58.2% | 70.6% | 71.3% | 64.2% |
| SHAR | 57.07% | 64.35% | 60.93% | 51.45% | 58.4% | 69.9% | 70.4% | 63.9% |
| HARQ-F | 49.15% | 55.86% | 52.64% | 42.34% | 58.7% | 70.3% | 71.0% | 65.4% |
| HExpGl | 55.64% | 63.00% | 55.21% | 31.63% | 57.8% | 70.2% | 70.6% | 63.1% |
| $OLS^{RM}$ | 50.25% | 55.28% | 45.75% | 25.64% | 59.8% | 71.4% | 71.6% | 64.3% |

Table 7 Robustness Analyses

This table reports the out-of-sample $R^2$ relative to the HAR model based on alternative structures of the Neural Network model in Panel A, different weighting schemes of the ensemble model across various forecast horizons in Panel B, 130 predictors including 118 main predictors, six cross-sectionally ranked firm characteristics, and six pure noise terms in Panel C, and 165 features including 63 lagged $RV$ and 102 lagged $IV$ features in Panel D. The sample consists of 173 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 100 index between January 1996 and June 2019 with share code 10 or 11, prices between \$1 and \$1000, daily number of trades greater than or equal to 100, and at least five years of data on all features and response variables. The features of each model specification consist of all 118 predictors detailed in Table 2. Panel A presents results for the Neural Network model with structures different from the baseline in Table 3. The full out-of-sample evaluation period is from January 2001 to June 2019 as in Table 3. Panel B presents the $R^2_{OOS}$ for the simple average of forecasts from the five individual machine learning models (AVG) and four ensemble models with different weighting schemes. The full out-of-sample evaluation period for all models in Panel B is from April 2002 to June 2019 because the last ensemble model $AVG^{ENet}$ requires 1-year data for training and 3-month data for validation. Column 2 provides a description of different NN structures and ensemble methods. $R^2_{OOS}$ for each model is calculated relative to the prediction from HAR using the entire panel of stocks according to Eq. (5).

*Panel A: Alternative structures for Neural Network*

| Model | Description | Daily | Weekly | Monthly | Quarterly |
|---|---|---|---|---|---|
| | | $R^2_{OOS}$ relative to HAR | | | |
| NN (Baseline) | # of hidden layer: 2; # of neurons: (5, 2) | 10.5% | 16.7% | 14.3% | 4.8% |
| NN1 | # of hidden layer: 1; # of neurons: (2) | 9.4% | 14.9% | 10.7% | 2.6% |
| NN2 | # of hidden layer: 2; # of neurons: (4, 2) | 10.5% | 17.4% | 12.6% | 6.0% |
| NN3 | # of hidden layer: 3; # of neurons: (8, 4, 2) | 8.3% | 15.0% | 13.2% | 5.2% |
| *Panel B: Alternative weighting schemes for the ensemble model* | | | | | |
| AVG | Simple average of forecasts from five individual machine learning models. | 9.8% | 14.5% | 14.2% | 8.6% |
| MED | Median of forecasts from five individual machine learning models. | 10.2% | 14.9% | 13.4% | 8.1% |
| $AVG^{Trim}$ | Simple average of forecasts from three out of five individual machine learning models after removing highest and lowest values. | 10.1% | 14.7% | 14.0% | 8.5% |
| $AVG^{ValidError}$ | Weighted average of forecasts from five individual machine learning models with weights equal to the inverse of validation MSE (normalized to sum up to one). | 9.8% | 14.7% | 14.2% | 8.7% |
| $AVG^{ENet}$ | Weighted average of forecasts from five individual machine learning models with weights tuned by Elastic Net. | 11.0% | 16.8% | 13.1% | 11.8% |
| *Panel C: Firm characteristics and pure noise features* | | | | | |
| AVG | 130 features: 16 $RV$ features + 102 $IV$ features + 6 firm characteristics + 6 pure noise terms | 9.1% | 14.5% | 15.2% | 9.7% |
| *Panel D: Raw lagged RV features* | | | | | |
| AVG | 165 features: 63 $RV$ features +102 $IV$ features | 7.6% | 12.8% | 14.1% | 10.6% |

Table 8 Out-of-sample predictions relative to HAR for S&P 500 stocks

This table reports the out-of-sample $R^2$ relative to the HAR model for OLS-based and ML-based volatility forecasting models across different forecast horizons for a different set of S&P 500 stocks. The sample consists of 663 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 500 index but not members of the S&P 100 index between January 1996 and June 2019 with share code 10 or 11, prices between \$1 and \$1000, daily number of trades greater than or equal to 100, and at least five years of data on all features and response variables. The full out-of-sample evaluation period is from January 2001 to June 2019. The features of each OLS-based model consist of either model-specific predictors or all 118 predictors as detailed in Table 2, and those of each ML-based model consist of all 118 predictors. Our ML-base models include LASSO, Principal Component Regression (PCR), Random Forest (RF), Gradient Boosted Regression Trees (GBRT), Neural Network (NN), and a simple average of forecasts from the five individual machine learning models (AVG). Hyperparameters for each ML-based model are provided in Table 3. We directly transfer the resulting tuning parameters for RF (i.e., maximum tree depth) and GBRT (i.e., # of trees and maximum tree depth) based on the original 173 S&P 100 stocks to this different set of 663 S&P 500 stocks and retrain both models without validating these tuning parameters. The remaining ML-based as well as OLS-based models are completely recalibrated using the new stock sample without hyperparameter transfer. $R^2_{OOS}$ for each model is calculated relative to the prediction from HAR using the entire panel of stocks according to Eq. (5).

| | Model | Tuning parameters transferred | Daily | Weekly | Monthly | Quarterly |
|---|---|---|---|---|---|---|
| | | | | $R^2_{OOS}$ *Relative to HAR* | | |
| | MIDAS | | 0.2% | 1.5% | 0.7% | -0.8% |
| | SHAR | | 0.8% | 1.0% | 0.8% | 0.4% |
| | HARQ-F | | 1.2% | 1.7% | 1.3% | 1.1% |
| OLS | HExpGl | | 0.5% | 2.0% | 1.7% | -0.3% |
| | $OLS^{RM}$ | | 3.3% | 5.2% | 3.4% | 0.5% |
| | $OLS^{IV}$ | | -15.1% | -20.3% | -18.4% | -13.6% |
| | $OLS^{ALL}$ | | 4.9% | 8.6% | 6.5% | 0.8% |
| | LASSO | | 5.0% | 9.1% | 7.8% | 2.2% |
| | PCR | | 4.3% | 7.6% | 3.6% | 4.5% |
| | RF | Maximum tree depth | 4.7% | 7.2% | 5.1% | 3.2% |
| ML | GBRT | # of trees & maximum tree depth | 5.1% | 9.1% | 5.6% | 1.7% |
| | NN | | 8.5% | 15.1% | 12.0% | 4.3% |
| | AVG | | 7.3% | 12.8% | 10.8% | 6.6% |

Table 9 Performance of variance risk premium strategy based on different $RV$ forecasts

This table reports the performance of the variance risk premium (VRP) strategy based on different $RV$ forecasts. The sample consists of 173 (836) stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 100 (S&P 500) index between January 1996 and June 2019 with share code 10 or 11, prices between \$1 and \$1000, daily number of trades greater than or equal to 100, and at least five years of data on all features and response variables. The full out-of-sample evaluation period is from January 2001 to June 2019. By the end of each day $t$, $VRP_t$ is measured as $IV_t - E_t(RV_{t+21})$; $IV_t$ is the at-the-money implied variance from call options with one-month maturity; $E_t(RV_{t+21})$ is the expected realized variance for day $t+21$ measured using the true value of the 21-day ahead realized variance $RV_{t+21}$ or the forecast from a given model. Panel A (B) reports the returns in monthly percentage of decile portfolios formed by VRP based on different $E_t(RV_{t+21})$ measures for S&P 100 (S&P 500) sample. By the end of day $t$, we sort stocks into ten portfolios by a given measure of $VRP_t$, and compute the value-weighted return of a self-financing portfolio that buys stocks in the top decile with high VRP and sells stocks in the bottom decile with low VRP with a 21-day holding period. The row labeled "High – Low" reports the difference in returns between Portfolio 10 and Portfolio 1, with Newey-West adjusted $t$-statistics in parentheses. The rows below report the alphas from the CAPM model, the Fama-French-3-factor model, or the Fama-French-5-factor model.

| | Panel A: S&P 100 | | | | Panel B: S&P 500 | | | |
|---|---|---|---|---|---|---|---|---|
| | $RV_{t+21}$ | $OLS^{ALL}$ | $NN$ | $AVG$ | $RV_{t+21}$ | $OLS^{ALL}$ | $NN$ | $AVG$ |
| 1 (Low) | -2.08 | 0.14 | 0.05 | 0.03 | -2.60 | -0.09 | -0.09 | -0.21 |
| 2 | -0.92 | 0.20 | 0.17 | 0.18 | -1.10 | 0.13 | 0.08 | 0.13 |
| 3 | -0.37 | 0.35 | 0.29 | 0.29 | -0.49 | 0.26 | 0.21 | 0.23 |
| 4 | 0.07 | 0.34 | 0.34 | 0.31 | 0.01 | 0.33 | 0.28 | 0.26 |
| 5 | 0.49 | 0.38 | 0.35 | 0.30 | 0.47 | 0.33 | 0.34 | 0.30 |
| 6 | 0.80 | 0.36 | 0.36 | 0.38 | 0.82 | 0.38 | 0.37 | 0.31 |
| 7 | 1.01 | 0.43 | 0.45 | 0.40 | 1.14 | 0.42 | 0.43 | 0.38 |
| 8 | 1.39 | 0.44 | 0.53 | 0.48 | 1.44 | 0.46 | 0.51 | 0.44 |
| 9 | 1.71 | 0.59 | 0.72 | 0.64 | 1.60 | 0.50 | 0.55 | 0.58 |
| 10 (High) | 1.80 | 0.53 | 0.60 | 0.74 | 1.65 | 0.34 | 0.35 | 0.51 |
| High – Low | 3.88 | 0.39 | 0.55 | 0.71 | 4.25 | 0.43 | 0.44 | 0.72 |
| | (11.92) | (2.21) | (2.98) | (3.72) | (12.82) | (2.40) | (2.25) | (3.54) |
| CAPM alpha | 3.89 | 0.38 | 0.55 | 0.70 | 4.26 | 0.42 | 0.44 | 0.71 |
| | (12.00) | (2.17) | (2.94) | (3.68) | (12.92) | (2.36) | (2.23) | (3.52) |
| FF3 alpha | 3.89 | 0.38 | 0.54 | 0.70 | 4.26 | 0.42 | 0.44 | 0.71 |
| | (12.01) | (2.18) | (2.94) | (3.69) | (12.95) | (2.37) | (2.23) | (3.53) |
| FF5 alpha | 3.87 | 0.40 | 0.56 | 0.71 | 4.24 | 0.44 | 0.45 | 0.72 |
| | (12.08) | (2.25) | (3.03) | (3.72) | (13.05) | (2.43) | (2.26) | (3.55) |

## References


Amihud, Y., 2002. Illiquidity and stock returns: Cross-Section and time-series effects. Journal of Financial Markets 5, 31–56.

Andersen, T. G., Bollerslev, T., 1998. Answering the skeptics: Yes, standard volatility models do provide accurate forecasts. International Economic Review 39, 885–905.

Andersen, T. G., Bollerslev, T., Christoffersen, P. F., Diebold, F. X., 2006. Volatility and correlation forecasting. In G. Elliott, C. W. J. Granger, and A. Timmermann, eds. Handbook of Economic Forecasting. North Holland, Amsterdam.

Andersen, T. G., Bollerslev, T., Diebold, F. X., 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., Bollerslev, T., Diebold, F. X., Labys, P., 2001. The distribution of realized exchange rate volatility. Journal of the American Statistical Association 96, 42–55.

Andersen, T. G., Bollerslev, T., Diebold, F. X., Labys, P., 2003. Modeling and forecasting realized volatility. Econometrica 71, 579–625.

Audrino, F., Knaus, S. D., 2016. Lassoing the HAR model: A model selection perspective on realized volatility dynamics. Econometric Review 35, 1485–1521.

Bali, T. G., Goyal, A., Huang, D., Jiang, F., Wen, Q., 2022. Predicting corporate bond returns: Merton meets machine learning. Working paper, Georgetown University, University of Lausanne, Swiss Finance Institute, Singapore Management University, and Central University of Finance and Economics.

Bao, Y., Jiang, X., 2016. An intelligent medicine recommender system framework. In: *2016 IEEE 11Th conference on industrial electronics and applications (ICIEA)*, IEEE, pp. 1383–1388.

Barndorff-Nielsen, O. E., Kinnebrock, S., Shephard, N., 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.

Barndorff-Nielsen, O. E., Shephard, N., 2002. Econometric analysis of realized volatility and its use in estimating stochastic volatility models. Journal of the Royal Statistical Society Series B 64, 253–280.

Bollerslev, T., Hood, B., Huss, J., Pedersen, L. H., 2018. Risk everywhere: Modeling and managing volatility. Review of Financial Studies 31, 2729–2773.

Bollerslev, T., Li, S. Z., Todorov, V., 2016a. Roughing up beta: Continuous versus discontinuous betas and the cross-section of expected stock returns. Journal of Financial Economics 120, 464–490.

Bollerslev, T., Li, S. Z., Zhao, B., 2020. Good volatility, bad volatility, and the cross section of stock returns. Journal of Financial and Quantitative Analysis 55, 751–781.

Bollerslev, T., Patton, A. J., Quaedvlieg, R., 2016b. Exploiting the errors: A simple approach for improved volatility forecasting. Journal of Econometrics 192, 1–18.

Bucci, A., 2020. Realized volatility forecasting with neural networks. Journal of Financial Econometrics 18, 502–531.

Busch, T., Christensen, B. J., Nielsen, M. O., 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.

Carr, P., Wu, L., Zhang, Z., 2020. Using machine learning to predict realized variance. Journal of Investment Management 18, 1–16.

Carvalho, C. M., Lopes, H. F., McCulloch, R. E., 2018. On the long-run volatility of stocks. Journal of the American Statistical Association 113, 1050–1069.

Chen, L., Pelger, M., Zhu, J., 2023. Deep learning in asset pricing. Management Science, forthcoming.

Christensen, B. J., Prabhala, N. R., 1998. The relation between implied and realized volatility. Journal of Financial Economics 50, 125–150.

Christoffersen, P., Goyenko, R., Jacobs, K., Karoui, M., 2018. Illiquidity premia in the equity options market. Review of Financial Studies 31, 811–851.

Connor, G., Korajczyk, R. A., Linton, O., 2006. The common and specific components of dynamic volatility. Journal of Econometrics 132, 231–255.

Corsi, F., 2009. A simple approximate long-memory model of realized volatility. Journal of Financial Econometrics 7, 174–196.

Da, R., Xiu, D., 2021. When moving-average models meet high-frequency data: Uniform inference on volatility. Econometrica 89, 2787–2825.

Diebold, F. X., Mariano, R. S., 1995. Comparing predictive accuracy. Journal of Business & Economic Statistics 13, 253–263.

Eraker, B., 2004. Do stock prices and volatility jump? Reconciling evidence from spot and option prices. Journal of Finance 59, 1367–1403.

Eraker, B., 2021. The volatility premium. Quarterly Journal of Finance 11, 1–35.

Eraker, B., Wang, J., 2015. A non-linear dynamic model of the variance risk premium. Journal of Econometrics 187, 547–556.

Fama, E. F., French, K. R., 1993. Common risk factors in the returns on stocks and bonds. Journal of Financial Economics 33, 3–56.

Fama, E. F., French, K. R., 2015. A five-factor asset pricing model. Journal of Financial Economics 116, 1–22.

Fisher, A., Rudin, C., Dominici, F., 2019. All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. Journal of Machine Learning Research 20, 1–81.

Ghysels, E., Santa-Clara, P., Valkanov, R., 2006. Predicting volatility: Getting the most out of return data sampled at different frequencies. Journal of Econometrics 131, 59–95.

Gu, S., Kelly, B., Xiu, D., 2020. Empirical asset pricing via machine learning. Review of Financial Studies 33, 2223–2273.

Guijaro-Ordonez, J., Pelger, M., Zanotti, G., 2022. Deep learning statistical arbitrage. Working paper, Stanford University.

Han, B., Zhou, Y., 2011. Variance risk premium and cross-section of stock returns. Working paper, University of Toronto and San Francisco State University.

Han, Y., Liu, F., Tang, X., 2020. The information content of the implied volatility surface: Can option prices predict jumps? Working paper, University of North Carolina at Charlotte, Cornell University, and University of Texas at Dallas.

Herskovic, B., Kelly, B., Lustig, H., Van Nieuwerburgh, S., 2016. The common factor in idiosyncratic volatility: Quantitative asset pricing implications. Journal of Financial Economics 119, 249–283.

Jiang, H., Li, S., Wang, H., 2021. Pervasive underreaction: Evidence from high-frequency data. Journal of Financial Economics 141, 573–599.

Jiang, J., Kelly, B., Xiu, D., 2022. (Re-)imag(in)ing price trends. Journal of Finance, forthcoming.

Kamara, A., Korajczyk, R. A., Lou, X., Sadka, R., 2016. Horizon pricing. Journal of Financial and Quantitative Analysis 51, 1769–1793.

Kaniel, R., Lin, Z., Pelger, M., Van Nieuwerburgh, S., 2022. Machine-learning the skill of mutual fund managers. Working paper, University of Rochester, Stanford University, and Columbia University.

Kelly, B., Pruitt, S., Su, Y., 2019. Characteristics are covariances: A unified model of risk and return. Journal of Financial Economics 134, 501–524.

Kingma, D. P., Ba, J., 2015. Adam: A method for stochastic optimization. 3rd International Conference for Learning Representations, Conference paper.

Li, B., Rossi, A., 2021. Selecting mutual funds from the stocks they hold: a machine learning approach. Working paper, Wuhan University and Georgetown University.

Liu, L., Patton, A. J., Sheppard, K., 2015. Does anything beat 5-minute RV? a comparison of realized measures across multiple asset classes. Journal of Econometrics 187, 293–311.

Lundberg, S. M., Lee, S.-I., 2017. A unified approach to interpreting model predictions. Advances in neural information processing systems 30.

Luong, C., Dokuchaev, N., 2018. Forecasting of realised volatility with the random forests algorithm. Journal of Risk and Financial Management 11, 1–15.

Murray, S., Xiao, H., Xia, Y., 2022. Charting by machines. Working paper, Georgia State University.

Newey, W. K., West, K. D., 1987. A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. Econometrica 55, 703–708.

Pan, S. J., Yang, Q., 2010. A survey on transfer learning. IEEE Transactions on knowledge and data engineering 22, 1345–1359.

Patton, A. J., Sheppard, K., 2015. Good volatility, bad volatility: Signed jumps and the persistence of volatility. Review of Economics and Statistics 97, 683–697.

Paye, B. S., 2012. 'Déjà vol': Predictive regressions for aggregate stock market volatility using macroeconomic variables. Journal of Financial Economics 106, 527–546.

Rahimikia, E., Poon, S.-H., 2023. Machine learning for realised volatility forecasting. Working paper, University of Manchester.

Rapach, D. E., Strauss, J. K., Zhou, G., 2013. International stock return predictability: What is the role of the United States? Journal of Finance 68, 1633–1662.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., Fei-Fei, L., 2015. Imagenet large scale visual recognition challenge. International journal of computer vision 115, 211–252.

Swanson, N. R., White, H., 1997. Forecasting economic time series using flexible versus fixed specification and linear versus nonlinear econometric models. International Journal of Forecasting 13, 439–461.

Taylor, S., 2007. Asset price dynamics, volatility, and prediction. Princeton, NJ: Princeton University Press.

Timmermann, A., 2006. Forecast combinations. In G. Elliott, C.W.J. Granger, and A. Timmermann, eds., Handbook of Economic Forecasting 1, 135–196.

Wolpert, D. H., 1996. The lack of a priori distinctions between learning algorithms. Neural Computation 8, 1341–1390.

Zhang, L., Mykland, P. A., Aït-Sahalia, Y., 2005. A tale of two time scales: Determining integrated volatility with noisy high-frequency data. Journal of the American Statistical Association 100, 1394–1411.

# Appendix of

# Automated Volatility Forecasting

## A.1. High-Frequency Data Cleaning

We begin by removing entries that satisfy at least one of the following criteria: a price less than or equal to zero; a trade size less than or equal to zero; corrected trades (i.e., trades with Correction Indicator, CORR, other than 0, 1, or 2); and an abnormal sale condition (i.e., trades for which the Sale Condition, COND, has a letter code other than @, *, E, F, @E, @F, *E, or *F). We then assign a single value to each variable for each second. If one or multiple transactions have occurred in that second, we calculate the sum of volumes, the sum of trades, and the volume-weighted average price within that second. If no transaction has occurred in that second, we enter zero for volume and trades. For the volume-weighted average price, we use the entry from the nearest previous second. Motivated by our analysis of the trading volume distribution across different exchanges over time, we purposely incorporate information from all exchanges covered by the TAQ database.

## A.2. Features

### *A.2.1. HAR*

The Heterogeneous Autoregressive (HAR) model proposed by Corsi (2009) is popular because it is easy to implement yet very effective in practice. The idea is to mix short- (daily), medium- (weekly), and long-term (monthly) volatility components for capturing various empirical properties observed in volatility series such as long memory and fat tails. The original HAR is used to forecast volatility up to monthly horizon. As our longest forecast horizon is quarterly, we augment the HAR model with a quarterly $RV$ term:

$$RV_{t+1}^{t+h} = \beta_0 + \beta_d RV_t^d + \beta_w RV_t^w + \beta_m RV_t^m + \beta_q RV_t^q + \epsilon_t, \tag{A.1}$$

where $RV_t^w$, $RV_t^m$ and $RV_t^q$ denote the average annualized daily $RV$ over lags 1 to 5, lags 1 to 21, and lags 1 to 63 throughout the paper.

### *A.2.2. MIDAS*

The mixed data sampling (MIDAS) model of Ghysels, Santa-Clara, and Valkanov (2006) assumes the following specification:

$$RV_{t+1}^{t+h} = \beta_0 + \beta_1 MIDAS_t^k + \epsilon_t, \tag{A.2}$$

in which the $MIDAS^k$ term is defined by:

$$\begin{aligned} MIDAS_t^k &= \frac{1}{\sum_{i=1}^{L} a_i}(a_1 RV_t^d + a_2 RV_{t-1}^d + \cdots + a_L RV_{t-L+1}^d), \\ a_i &= (\frac{i}{L})^{\theta_1 - 1}(1 - \frac{i}{L})^{\theta_2 - 1}\Gamma(\theta_1 + \theta_2)\Gamma(\theta_1)^{-1}\Gamma(\theta_2)^{-1}, \quad i = 1, ..., L, \end{aligned} \tag{A.3}$$

where $\Gamma(\cdot)$ denotes the Gamma function; the superscript $k$ in $MIDAS^k$ can take values of $d, w, m, q$, representing the resulting $MIDAS$ term from predicting $h = 1,\ 5,\ 21,\ 63$-day-ahead $RV$. The $MIDAS$ feature can be viewed as a smoothly weighted sum of lagged daily $RV$'s. It has three hyperparameters $\theta_1$, $\theta_2$, and $L$ that need to be tuned. Directly mirroring Ghysels, Santa-Clara, and Valkanov (2006) and Bollerslev, Hood, Huss, and Pedersen (2018), we set $\theta_1 = 1$ and $L = 50$. Further guided by Bollerslev, Hood, Huss, and Pedersen (2018), we employ a grid search to tune $\theta_2$ for each $h$-day forecast horizon and choose the value that minimizes the Mean Squared Errors (MSE) over the full sample.[1]

### *A.2.3. SHAR*

We follow Patton and Sheppard (2015) to estimate a Semivariance-HAR (SHAR) model that decomposes daily $RV$ into two realized semivariance components:

$$RV_{t+1}^{t+h} = \beta_0 + \beta_d^+ RVP_t^d + \beta_d^- RVN_t^d + \beta_w RV_t^w + \beta_m RV_t^m + \beta_q RV_t^q + \epsilon_t, \tag{A.4}$$

[1] To avoid onerous computational burdens, we follow the literature and do not perform a rolling grid search for the $\theta_2$ parameter. As a result, the MIDAS feature is not truly out-of-sample but is included for comparison.

where the annualized daily positive and negative semivariances, introduced by Barndorff-Nielsen, Kinnebrock, and Shephard (2010), are defined as:

$$RVP_t^d = 252 \times \sum_{i=1}^{n} r_{t-1+i/n}^2 1_{\{r_{t-1+i/n}>0\}},\ RVN_t^d = 252 \times \sum_{i=1}^{n} r_{t-1+i/n}^2 1_{\{r_{t-1+i/n}<0\}}. \tag{A.5}$$

Daily realized semivariances provide a natural decomposition of daily $RV$, i.e., $RV_t^d = RVP_t^d + RVN_t^d$. Patton and Sheppard (2015) show that the negative semivariance $RVN^d$ has stronger predictive power on future $RV$'s.[2] To mitigate bias in realized semivariance estimates, we also apply the subsampling scheme to construct $RVP^d$ and $RVN^d$.

### *A.2.4. HARQ-F*

Bollerslev, Patton, and Quaedvlieg (2016b) propose a HARQ-F model by considering measurement errors in $RV$ estimates. The measurement error may be characterized by the asymptotic (for $n \to \infty$) distribution theory of Barndorff-Nielsen and Shephard (2002):

$$RV_t = IV_t^* + \epsilon_t,\ \ \epsilon_t \sim MN(0, 2\Delta IQ_t), \tag{A.6}$$

where $IV_t^* \equiv \int_{t-1}^{t} \sigma_s^2 ds$ is the unobservable Integrated Variance, $IQ_t \equiv \int_{t-1}^{t} \sigma_s^4 ds$ denotes the Integrated Quarticity ($IQ$), and MN stands for mixed normal. Using intraday returns, the integrated quarticity for *annualized* daily $RV$ may be consistently estimated by annualized daily realized quarticity ($RQ$):

$$RQ_t^d = 252^2 \times \frac{n}{3} \sum_{i=1}^{n} r_{t-1+i/n}^4. \tag{A.7}$$

To improve efficiency, we further apply the subsampling method to the daily $RQ$ estimation. Weekly, monthly, and quarterly realized quarticities, denoted by $RQ^w$, $RQ^m$ and $RQ^q$, respectively, can be calculated by averaging daily $RQ$ over lags 1 to 5, lags 1 to 21, and lags 1 to 63. The HARQ-F

[2]Patton and Sheppard (2015) also rely on the difference between positive and negative realized semivariances to isolate signed jumps $\Delta J_t^2 = RVP_t^d - RVN_t^d$, and show that $\Delta J_t^2$ negatively predicts future $RV$. Our Eq. (A.4) nests the specification of including $\Delta J_t^2$ when $\beta_d^+ = -\beta_d^-$. On a related note, Andersen, Bollerslev, and Diebold (2007) find that unsigned jumps lead to only a slight decrease in future $RV$.

model allows coefficients of lagged $RV$'s to vary as a function of $\sqrt{RQ}$:

$$\begin{aligned} RV_{t+1}^{t+h} = \beta_0 + \beta_d RV_t^d + \beta_w RV_t^w + \beta_m RV_t^m + \beta_q RV_t^q \\ + \phi_d RV_t^d \sqrt{RQ_t^d} + \phi_w RV_t^w \sqrt{RQ_t^w} + \phi_m RV_t^m \sqrt{RQ_t^m} + \phi_q RV_t^q \sqrt{RQ_t^q} + \epsilon_t. \end{aligned} \tag{A.8}$$

Bollerslev, Patton, and Quaedvlieg (2016b) show that, by allowing the model parameters to vary explicitly with the degree of measurement error, this model generates significant improvements in the accuracy of the forecasts compared with the forecasts from some of the most popular risk models.

### *A.2.5. HExpGl*

The Heterogeneous Exponential Realized Volatility with Global Risk Factor (HExpGl) model by Bollerslev, Hood, Huss, and Pedersen (2018) represents one of the latest techniques for volatility forecasting. Like HAR and MIDAS, HExpGl also constructs features based on daily $RV$ series. The difference is that HExpGl uses exponentially weighted moving averages (EWMA) of lagged daily $RV$'s, whereas HAR uses step functions and MIDAS relies on more complicated functional forms. The EWMA of lagged daily $RV$'s with a pre-specified center-of-mass ($CoM$) is given by:

$$ExpRV_t^{CoM(\lambda)} = \sum_{i=1}^{500} \frac{e^{-i\lambda}}{e^{-\lambda} + e^{-2\lambda} + \cdots + e^{-500\lambda}} RV_{t-i+1}^d, \tag{A.9}$$

where $\lambda$ defines the decay rate of the weights and $CoM(\lambda)$ denotes the corresponding center-of-mass $CoM(\lambda) = e^{-\lambda}/(1 - e^{-\lambda})$; conversely, for a given center-of-mass, $\lambda$ can be inferred from $\lambda = log(1 + 1/CoM)$. The center-of-mass for a given $ExpRV$ measure captures the "average" horizon of the lagged $RV$'s that it uses. We follow Bollerslev, Hood, Huss, and Pedersen (2018) to consider $ExpRV$ terms with center-of-mass equal to 1, 5, 25, and 125 trading days. Motivated by the cross-asset and cross-market volatility spillover effects, HExpGl also includes the EWMA of a global risk factor $GlRV$ with a center-of-mass equal to 5:

$$ExpGlRV_t^5 = \sum_{i=1}^{500} \frac{e^{-i\lambda}}{e^{-\lambda} + e^{-2\lambda} + \cdots + e^{-500\lambda}} GlRV_{t-i+1}, \tag{A.10}$$

where the corresponding $\lambda = log(1 + 1/CoM) = log(1 + 1/5)$. For each day $t$ and each stock $i$, the global risk factor $GlRV$ is computed as the average normalized $RV$ scaled back to the asset's own long-run mean of $RV$, that is, $(\frac{1}{N}\sum_{j=1}^{N}\frac{RV_{j,t}^{d}}{\overline{RV_j}})\overline{RV_i}$, where $\overline{RV_i}$ is the long-run mean of daily $RV$ for stock $i$ calculated from the beginning of the sample period until day $t$. The resulting HExpGl model specification is given by:

$$\begin{aligned} RV_{t+1}^{t+h} = \beta_0 + \beta_1 ExpRV_t^1 + \beta_2 ExpRV_t^5 + \beta_3 ExpRV_t^{25} + \beta_4 ExpRV_t^{125} \\ + \beta_5 ExpGlRV_t^5 + \epsilon_t. \end{aligned} \tag{A.11}$$

### *A.2.6. Option-Implied Variances*

In addition to the high-frequency-based realized features from the existing models, our paper also considers option-implied variances as inputs. Because our forecasting horizon is up to three months, we include all 102 individual stock options from put and call options with maturities between one and three months across all deltas to avoid cherry-picking a particular option in order to reduce the chance of overfitting. For call-option-implied variances, we denote these features as $CIV^{jm,\delta}$ with maturity equal to $j$ months ($j = 1, 2, 3$) and delta equal to $\delta$ ($\delta = 0.1, 0.15, ..., 0.9$). For put-option-implied variances, we denote these features as $PIV^{jm,\delta}$ with maturity equal to $j$ months ($j = 1, 2, 3$) and delta equal to $\delta$ ($\delta = -0.1, -0.15, ..., -0.9$).

## A.3. Machine Learning Algorithms

### *A.3.1. LASSO*

LASSO is designed to improve performance over that of OLS by imposing sparsity-encouraging penalties on regression coefficients for variance reduction and model interpretation. Take daily $RV$ prediction as an example, LASSO assumes the same linear regression function as OLS:

$$g(z_{i,t};\theta) = z_{i,t}'\theta, \tag{A.12}$$

where $z'_{i,t}$ is the feature vector for stock $i$ on day $t$ and $\theta$ is the unknown parameter. Unlike OLS, however, LASSO estimates $\theta$ through a penalized $L_1$ loss function:

$$\mathcal{L}(\theta;\lambda) = \frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}(RV_{i,t+1}^{d} - g(z_{i,t};\theta))^2 + \lambda\sum_{j=1}^{P}|\theta_j|, \tag{A.13}$$

where $\lambda$ is the shrinkage parameter that controls for the magnitude of the penalty on the coefficients. The special case of $\lambda = 0$ collapses back to OLS. In such a case, LASSO/OLS minimizes the training (in-sample) error, potentially overfitting the data. By imposing the $L_1$ penalty $\lambda\sum_{j=1}^{P}|\theta_j|$, LASSO is capable of setting some of the coefficients to be exactly zero, a very desirable property for two reasons. First, setting coefficients to zero reduces parameter estimation variance and thus brings down the variance component of the prediction error. Second, with zero regression coefficients, the fitted model becomes more interpretable.

It is important to consider several implementation details to achieve better performance with LASSO. First, we need to normalize features before estimating the models so that all features have comparable magnitudes. Otherwise, a single $\lambda$ would have vastly different shrinkage effects on different features, making it impossible to tune. The normalization is done by using only mean and standard deviation of the training sample to prevent look-ahead bias; we recalculate the mean and standard deviation once per year to be consistent with the expanding window scheme detailed in Section 3.1. Second, we need to choose $\lambda$ from a wide range of values that can generate coefficient estimates with varying sparsity levels for the model selection procedure to choose from. Otherwise, the selected $\theta$ might be far from the region of optimal fit in the parameter space.

### *A.3.2. Principal Component Regression*

The second linear learning algorithm we consider is PCR, which is motivated by the fact that our volatility forecasting features are often correlated. PCR uses dimension-reduction techniques to produce a small number of common factors from the original feature space and then relies on the derived features as inputs for regressions. Specifically, in the first step, Principal Component Analysis (PCA) is performed on the $P$-dimensional original feature space to extract a small number of factors as linear combinations of the original inputs; these factors are orthogonal to each other to prevent information redundancy. In the second step, we take only the first $K$ most important

principals that preserve the main variability of the original features for fitting the regression. More formally, PCR is defined as follows:

$$RV = (Z\Omega_K)\theta_K + \tilde{E}, \tag{A.14}$$

where $RV$ is the $NT \times 1$ vector of realized variances, $Z$ is the $NT \times P$ matrix of features, $\Omega_K$ is a $P \times K$ orthogonal projection matrix from the $P$-dimensional original feature space onto the $K$-dimensional derived input space, $\theta_K$ is a vector of coefficients corresponding to $K$ derived inputs, and $\tilde{E}$ is an $NT \times 1$ vector of residuals. The projection matrix $\Omega_K$ can be found through singular value decomposition (SVD) of the original feature matrix $Z$.

The hyperparameter for PCR is the number of derived input features $K$. There is a trade-off between dimension reduction and information preservation when choosing $K$. If $K$ is large, more information in the original features is kept and used to make predictions. Overfitting concerns naturally arise, however, as there are more parameters to estimate. If $K$ is small, there is a risk that the second-stage regression model misses some useful information in the discarded principal components. In our implementation, we choose $K$ through validation. This gives the unsupervised learning PCA some guidance based on the target. We also standardize all features, as in LASSO, to ensure the principal components are not dominated by a single feature with extremely large variance. The number of components used in the linear regression is chosen by the smallest MSE on the validation sets. To increase computational speed and also prevent overfitting, we set an upper bound for $K$ equal to 20.

### *A.3.3. Random Forest*

Our first nonlinear learning algorithm is the random forest (RF) model, which is based on regression trees for modeling nonlinearity. Unlike linear methods reviewed in the previous two sections that essentially project the response onto the feature space, tree-based models partition the feature space into a set of non-overlapping regions as illustrated in Figure A.1. The observations within the same region are then fit through a simple model such as a constant. Mathematically, the estimated

response function of a regression tree is:

$$\widehat{g}(z^*_{i,t};\theta,K,L)=\sum_{k=1}^{K}\theta_k 1_{\{z^*_{i,t}\in C_k(L)\}}, \tag{A.15}$$

where $C_k(L)$ is one of the $K$ regions pre-determined by the training set. $K$ is the number of regions, $L$ is the tree depth, $1_{\{\cdot\}}$ is an indicator function, and $\theta_k$ is the sample mean of the outcomes for training observations within that region. A very large tree with many regions can capture very fine details of the data but is prone to overfitting. Consider the extreme case where a fully grown tree divides every single training observation in the training set into one region, thus yielding zero training error but very poor out-of-sample performance.

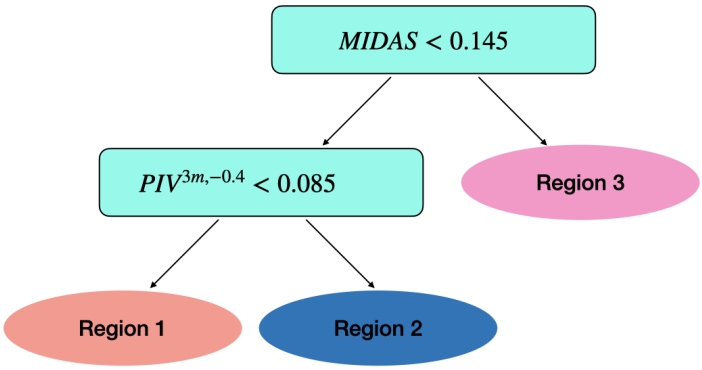



Fig. A.1 Illustration of a regression tree model

RF reduces the overfitting problem associated with regression trees through several modifications. First, instead of a single tree, RF generates multiple trees by bootstrapping the training sample and then averaging forecasts from each individual tree to reduce the variance. Second, RF implements observation and feature subsampling in the training process to decorrelate individual trees in the forest for further variance reduction.

How large should we grow the trees? As described earlier, deep trees are less biased but very unstable. Our strategy is to grow a large tree and then prune it back to a depth of $L$. We tune the tree depth $L$ via validation where we search for the optimal $L$ that minimizes the validation error over a grid of values ranging from 1 to 20. For each RF fitting, we bootstrap and average over 500 trees. For each tree, we use 50% of the training observations, and for each node split, we use $ln(P)$ features. Tree-based models are insensitive to feature location and scale and thus do not require

feature standardization.

### *A.3.4. Gradient Boosted Regression Trees*

The second nonlinear learning algorithm we investigate is the Gradient Boosted Regression Trees (GBRT), which uses the base learner of regression trees as RF. There are, however, two principal differences between GBRT and RF. First, GBRT uses trees as base learners in an *additive* fashion whereas RF uses trees in an *average* fashion. At each step, GBRT fits a new tree to explain what has been left unexplained by previous trees, while RF fits a parallel tree to explain the original response. Second, GBRT prefers using shallow trees because each tree is supposed to be weak, but by adding many small trees GBRT gradually reduces prediction bias while still controlling for variance. In contrast, RF prefers deep trees because these trees need to be unbiased, and only by averaging many deep trees is RF expected to reduce variance while simultaneously capturing the true relation.

To prevent overfitting, GBRT adds a new tree after discounting its contribution. Specifically, at every round after fitting a tree to the residuals, we update our $\widehat{g}(\cdot)$ by adding a shrunken version of the new tree with a shrinkage multiplier $0 < \lambda < 1$, which is called the learning rate. We then update the residuals by subtracting this shrunken tree from the previously predicted values. Other approaches employed by RF to mitigate overfitting problems are also used for GBRT. Specifically, we adopt subsampling for each tree and randomly draw a subset of features at each split. The hyperparameters for GBRT are the learning rate $\lambda$ which controls the speed of learning, the maximum tree depth that represents the upper bound for the degree of polynomials and interactions, and the number of trees which prevents overfitting and as a result can balance the in-sample performance with the out-of-sample prediction.

In our implementation, we set the learning rate $\lambda$ low at 0.001 to help prevent the model from overfitting the residuals. We validate the maximum tree depth, $L$, from 1 to 5. The grids with $L > 1$ are set to give GBRT the ability to include high-order interactions and polynomials. For subsampling, we again use 50% of the training observations for each tree and $ln(P)$ features for each split. In addition, we use early-stopping rules to help us choose the number of trees in the GBRT model: 1) If Mean Squared Errors (MSE) stop decreasing after 50 consecutive rounds, we set the number of trees as the round at which the MSE stops improving instead of including more trees in our GBRT model, and 2) when the total number of trees reaches 20,000. We report the

resulting number of trees as model complexity. Like RF, GBRT is location- and scale-invariant so there is no feature standardization.

### *A.3.5. Feed-Forward Neural Network*

Our third nonlinear model is the feed-forward Neural Network (NN), which uses hidden layers and nonlinear transformations to capture complex nonlinear relations. As shown in Figure A.2, the original inputs $X$ pass through one or more hidden layers, which transform these inputs into derived features $Z$. The output layer aggregates the derived features into the final prediction. Transformations are called activation functions in NN and are the sources of nonlinearity.

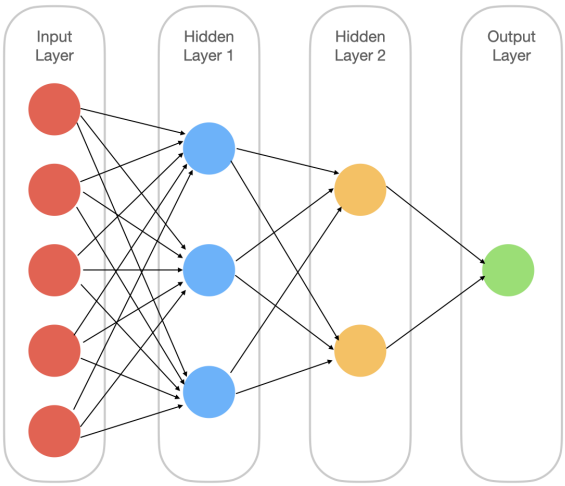



Fig. A.2 Illustration of a feed-forward neural network model

In our implementation, we consider a model that has two hidden layers with 5 and 2 neurons, respectively. For the activation function, we choose the commonly used rectified linear unit (ReLU) given as:

$$ReLU(x) = max(x, 0). \tag{A.16}$$

We solve for the parameters in the activation function via stochastic gradient descent (SDG). We choose the adaptive moment estimation (Adam) by Kingma and Ba (2015) for computational efficiency and standardize each feature because NN is sensitive to feature scales. We also use multiple random states when implementing stochastic optimization for hyperparameters and derive predictions by averaging forecasts based on all tuned neural network models with ten starting points

in order to obtain reliable estimates.

## A.4. Additional Results

### *A.4.1. Factor Structures in Realized Variances*

In the extreme case when a common volatility factor fully captures the co-movements in cross-sectional $RV$'s, the dependency structure between the different $RV$ time series is nearly perfect. As a result, using panel data instead of time-series data would not improve model performance because the effective sample size stays roughly the same. However, Section 4.4 provides initial evidence that panel data produce more accurate volatility forecasts, indicating that the extreme case does not hold in practice. In this section, we conduct two additional analyses to shed light on how strong the factor structure in volatilities is and whether exploiting such factor structure can enhance volatility forecasting performance.

First, we perform a simple PCA on the covariance matrix based on the volatility time series of all stocks with full sample between 1996 and 2019, and find that the first principle component (PC) explains 33%, 41%, 49%, and 57% variation in the original data for daily, weekly, monthly, and quarterly $RV$'s. These numbers increase to 51%, 61%, 73%, and 84% when we use the first three PCs. Thus, despite the factor structure in volatilities, there is still substantial amount of variation in the data not captured by a small number of uncorrelated volatility factor time series. More importantly, the information not captured by common factors is proven to be highly useful in improving the volatility forecasting accuracy as shown in Table 6.

Second, following Herskovic, Kelly, Lustig, and Van Nieuwerburgh (2016), we construct a common volatility factor $CRV$ for each measurement horizon as the equal-weighted average across all $RV$s and run separate time-series regressions by regressing $RV$ of each stock on $CRV$. The cross-sectional average regression $R^2$'s for daily, weekly, monthly, and quarterly $RV$'s are 40.0%, 48.3%, 55.6%, and 59.5%, respectively, implying that about of the variations are not explained by the common volatility factor. Then for each measurement horizon, we decompose total $RV_{i,t}$ for stock $i$ on day $t$ into two components: (1) the component explained by the common factor, or the product of $CRV$ and its factor exposure, and (2) the residual component $RRV$ defined as the

difference between total $RV$ and the first component.[3] Next, we separately forecast $CRV$ and the residual components using various models.[4] Lastly, we rely on the forecasts of $CRV$ and $RRV$ to produce the final forecast of total $RV$. The resulting volatility prediction is poor no matter which models we use to forecast $CRV$ and $RRV$, and how we estimate $CRV$ exposures. There are potentially several reasons to explain why such attempt fails, including estimation errors in $CRV$ exposures, model misspefication, and the difficulty in forecasting $CRV$ using aggregate time-series data.

### *A.4.2. Firm Characteristics and Pure Noise Features*

Our 118 features are all volatility-based features and, given the persistence of volatility, they are naturally strong predictors of future $RV$'s. One may wonder about how our new system can handle other features such as weak features or even pure noise features. To address these questions, we consider two new feature sets: firm characteristics and pure random noises. In the volatility forecasting literature, firm characteristics have not been widely documented as useful predictors of future realized volatility.[5] On the other hand, it might be reasonable to hypothesize that firm characteristics such as size might be indirectly (through interaction or nonlinearity) helpful in volatility forecasting. To examine the power of firm characteristics, we consider the following features: $Size$, $BM$, $Mom$, $Ret^d$, $Ret^m$, and $ILLQ$. $Size$ is the natural logarithm of the product of the closing price and the number of shares outstanding by the end of the previous month from CRSP. $BM$ in June of year $t$ is computed as the ratio of the book value of common equity in fiscal year $t-1$ to the market value of equity in December in year $t-1$ and is updated every year. $Mom$ is the past 2 to 12 month cumulative returns. $Ret^d$ and $Ret^m$ refer to the past daily and monthly returns. $ILLQ$ is the illiquidity measure of Amihud (2002), which is the average daily ratio of the absolute stock return to the dollar trading volume over the previous month. Following Kelly, Pruitt, and Su (2019) and Gu, Kelly, and Xiu (2020), we cross-sectionally rank each characteristic on each day and map these ranks into the [-1,1] interval. Then we use the relative ranks of these

[3] Factor exposures are estimated by the regression coefficient using data prior to day $t$.

[4] When predicting $CRV$, we use all or subsets of 118 aggregate features that are cross-sectional averages of firm-level features. When predicting $RRV$, we use all or subsets of 118 residual features, each of which is constructed using the difference between the original feature and the product of $CRV$ exposure and the corresponding aggregate features.

[5] Paye (2012) shows that volatility forecasts exploiting macroeconomic variables do not outperform a univariate benchmark out-of-sample much, and Rahimikia and Poon (2023) find that adding news sentiment variables only marginally improves the forecasting performance.

characteristics as additional features.[6]

The second new feature set is pure noise, with which we can test how well our system handles false positives. We generate six random noise terms that mimic the distributional properties of the volatility-based features. Let $r_{i,j,t}$ denote the $j$-th noise term for stock $i$ on day $t$. We simulate the panel of noises for each $1 \leq i \leq N$ and each $1 \leq j \leq 6$ from the following model:

$$r_{i,j,t} = 0.2(1-\rho_j) + \rho_j r_{i,j,t-1} + u_{i,j,t}, \;\; u_{i,j,t} \sim \mathcal{N}(0, 0.25^2(1-\rho_j^2)), \tag{A.17}$$

where $\rho_j \in \{0.2, 0.4, 0.6, 0.8, 0.9, 0.99\}$ is the first-order autocorrelation of noise $j$. By construction, each noise term will have a mean of 0.2 and a standard deviation of 0.25, and they cover a wide range of persistence levels.

Panel B of Table A.4 presents the $R^2_{OOS}$'s relative to HAR for fitting $OLS^{ALL}$ and the ML-based models to the newly expanded set of 130 features. In Panel A of the same table we replicate the results reported in Table 3 using the original 118 features for ease of comparison. Overall, the augmented feature set generates very similar results to these using the original 118 features across different models. For $OLS^{ALL}$, the additional features maintain the same relative $R^2_{OOS}$'s at daily horizon, but reduce the relative $R^2_{OOS}$'s at weekly, monthly, and quarterly horizons as a result of overfitting more predictors. For LASSO, PCR, RF, and GBRT, the average performance of each model over forecast horizons stays about the same using either 118 or 130 features. For NN, the additional features produce similar relative $R^2_{OOS}$'s at the daily and weekly horizons, but deliver worse performance at the monthly and quarterly horizons. For the ensemble model AVG, the additional features show minimal improvement in the relative $R^2_{OOS}$'s at the first two horizons, identical relative $R^2_{OOS}$ at the monthly horizon, and slightly worse relative $R^2_{OOS}$ at the quarterly horizon.

Figure A.4 displays the group importance plots based on all 130 features for each individual ML model and the ensemble model AVG. In addition to the three groups of features from the original 118-feature set, we include two new groups, "Firm Char" and "Noise," each of which contains six cross-sectionally ranked firm characteristics and six pure noise terms, respectively. There are several intriguing observations. First, the importance of the first three groups is largely

[6] Using raw characteristics without transformation produces quantitatively similar results.

aligned with what Figure 3 shows based on 118 features. Secondly, cross-sectionally ranked firm characteristics as a group contributes modestly to $RV$ prediction, with group importance ranging from 0.03% for LASSO at the quarterly horizon to 4.07% for RF at the same quarterly horizon. Across models, the contribution of cross-sectionally ranked firm characteristics is relatively greater for nonlinear models RF, GBRT, and NN at around 2% at various forecast horizons compared to around 0.2% for linear models LASSO and PCR. One possible explanation is that cross-sectionally ranked firm characteristics can help predict future $RV$ only as interaction terms with existing $RV$ predictors. Lastly, the noise features contribute almost nothing to model prediction, indicating that our ML-based models and the associated group importance metrics effectively control for false positives.

### *A.4.3. Best Subset Selection Analysis and $R^2_{OOS}$ Decomposition*

We investigate the relative importance of feature groups for the performance of each ML model across time using permutation-based group importance measures in Section 4.6. One drawback of this method is that the model is only trained once at the full model and even with the permutation, the effect of the group of interest is never truly removed from the model. Alternatively, we consider the best subset selection approach that only fits a subset of features at a time and compare the OOS performance under different subsets.

Our analysis involves six subsets of three feature groups in total, comprising three one-group subsets and three two-group subsets. By utilizing these subsets, we retrain our automated system and calculate the relative $R^2_{OOS}$ for each subset to evaluate its contributions. To save computational resources and keep the presentation manageable, we focus on the monthly forecast horizon. Panel A of Figure 5 displays the relative $R^2_{OOS}$'s obtained from our sub-group analysis across feature group subsets. The plot shows that when utilizing only one feature group, the "RV & RQ" group achieves the highest relative $R^2_{OOS}$'s for our system. The $IV$ features, on the other hand, contribute least to $RV$ forecasting when fitted solely in the AVG model, which aligns with our previous analysis using only $IV$s with OLS-based models. Despite the $IV$ features being the least informative group, the positive relative $R^2_{OOS}$ of 3.8% achieved by the AVG-IV model suggests that forecast combination can reduce forecast variance and thus enhance prediction accuracy over OLS-based models. The relative $R^2_{OOS}$'s of our system based on two groups of features deliver a different message. $IV$s are

selected by our system as a complementary information set to either "MIDAS & ExpRV" or "RV & RQ" groups, resulting in $R^2_{OOS}$ improvements ranging from 6.2% to 7.7% when transitioning from one-group to two-group fittings. Finally, when all groups of features are included, the AVG model experiences further improvements in relative $R^2_{OOS}$'s and achieves the best performance across feature groups.

The results of the subgroup analysis reveal distinct patterns in the importance of feature groups. Specifically, "MIDAS & ExpRV" and "RV & RQ" terms demonstrate direct contributions to the predictions, while the $IV$ features offer interactive contributions. To capture both the individual and interactive importance of the groups, we adopt a similar but simplified approach inspired by the SHAP measure introduced by Lundberg and Lee (2017). This approach involves decomposing the total $R^2_{OOS}$'s into group-specific $R^2_{OOS}$'s using out-of-sample subgroup predictions within a coalitional game framework. Formally, the alternative importance measure for group $k$ is defined as:

$$\phi(k) = \sum_{S \subseteq \{F \setminus k\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} [R^2_{OOS}(S \cup k) - R^2_{OOS}(S)], \tag{A.18}$$

where $S$ represents a subset of the feature groups used in the model, $F$ denotes the set of all feature groups, $k$ is the target feature group, $|\cdot|$ indicates the number of groups in a feature set, and $R^2_{OOS}(\cdot, \cdot)$ is the relative $R^2_{OOS}$ for AVG model fitted using a particular feature group set. Accordingly, the importance of each feature group is a weighted average of all differences in $R^2_{OOS}$'s observed when the AVG model is trained with feature group $k$ presented v.s. withheld. To illustrate this calculation, let's consider the $IV$ group as an example. The group importance of the $IV$ features can be computed as follows:

$$\begin{aligned}
\phi(IV) =& \frac{1}{3}[R^2_{OOS}(IV) - 0] + \frac{1}{6}[R^2_{OOS}(MIDAS\&ExpRV + IV) - R^2_{OOS}(MIDAS\&ExpRV)] \\
&+ \frac{1}{6}[R^2_{OOS}(RV\&RQ + IV) - R^2_{OOS}(RV\&RQ)] \\
&+ \frac{1}{3}[R^2_{OOS}(MIDAS\&ExpRV + RV\&RQ + IV) - R^2_{OOS}(MIDAS\&ExpRV + RV\&RQ)],
\end{aligned}$$

This group importance measure possesses three desirable properties: 1) it considers both individual and interactive contributions, 2) it is grounded in out-of-sample performance, and 3) the sum of

group importance corresponds to the total $R^2_{OOS}$. Contrasting the permutation measure discussed in the main results, which primarily focus on the in-sample marginal dependence of models on feature groups, the new measure assesses the relative contributions of groups to out-of-sample forecast accuracy, thereby reflecting the model's performance beyond its specification. Panel B of Figure 5 reports the decomposition of total $R^2_{OOS}$'s under our automated system. Once again, we confirm that all three feature groups significantly contribute to the superior out-of-sample forecasts generated by our automated system, reaffirming the robustness of our system in forecast performance and model interpretation.

### *A.4.4. Firm Characteristics and Volatility Predictability*

To understand which characteristics are associated with greater cross-sectional heterogeneity in *RV* predictability, we employ simple portfolio sorts on the S&P 100 stock universe as follows. By the end of each month for a given forecast horizon, we sort all stocks into quintile portfolios by a given firm characteristic and compute for each portfolio the $R^2_{OOS}$ relative to HAR based on model AVG with 118 features in the following month. Table A.5 reports the time-series mean of the $R^2_{OOS}$'s for each quintile portfolio and the difference of the means between the highest and lowest quintile portfolios, with Newey-West robust $t$-statistic in parentheses. Across forecast horizons, $Size$ generates significantly positive $R^2_{OOS}$ spread, whereas $BM$ and $ILLQ$ produces significantly negative $R^2_{OOS}$ spread, indicating that large firms, growth firms, and more liquid firms in our sample are associated with stronger *RV* predictability. On the other hand, return-based characteristics $Mom$, $Ret^d$, and $Ret^m$ generally do not produce significant $R^2_{OOS}$ spread across horizons, indicating that volatility forecasting performance is not typically sensitive to price trend.

### *A.4.5. Robustness: Stock Universe and Sample Selection*

The empirical analyses discussed in our main results are based on the full history between January 1996 and June 2019 for stocks with at least five years of data. To demonstrate that the superior performance remains intact for any potential selection biases, we conduct additional evaluations of the out-of-sample performance. First, we relax the five-year requirement for stock inclusion in the sample used for model training and forecasting. Second, we exclude forecasts for the testing period when a stock has not yet entered the index.

To examine the potential selection bias arising from the five-year data requirement, we re-train our models using a new stock sample consisting of both the original S&P 100 stocks and S&P 100 stocks without five-year data. We then apply these retrained models to the new sample and observe consistent improvements in our $RV$ forecasts. Panel A of Table A.6 reports the relative $R^2_{OOS}$'s across forecasting models and horizons for the new S&P 100 sample. Nearly all models benefit from the enlarged sample sizes, particularly in longer forecast horizons, as evidenced by increases in the relative $R^2_{OOS}$'s across models. AVG, as expected, outperforms most of the other forecast models across four horizons, with $R^2_{OOS}$'s between 9.1% to 15.5%. These findings reaffirm that our automated system consistently produces the most accurate forecasts across different horizons and samples.

The out-of-sample performance, excluding forecasts made prior to a stock's inclusion in the S&P 500 index, is summarized in Panel B of Table A.6. In line with the earlier findings, the restricted testing sample yields very similar results to the original sample across different models. For $OLS^{ALL}$, the exclusion maintains or even increases the relative $R^2_{OOS}$'s at daily and weekly horizons, but reduces the relative $R^2_{OOS}$'s at the monthly and quarterly horizons. This suggests that a portion of the superior performance observed in the dense linear model is sensitive to the effective sample size. In contrast, AVG again demonstrates extraordinary out-of-sample performance. The relative $R^2_{OOS}$'s of AVG, using the restricted testing sample, range from 7.5% to 13%. These values slightly surpass the performance of AVG using the original testing sample, highlighting the robustness of forecast combinations in forecasting $RV$'s.

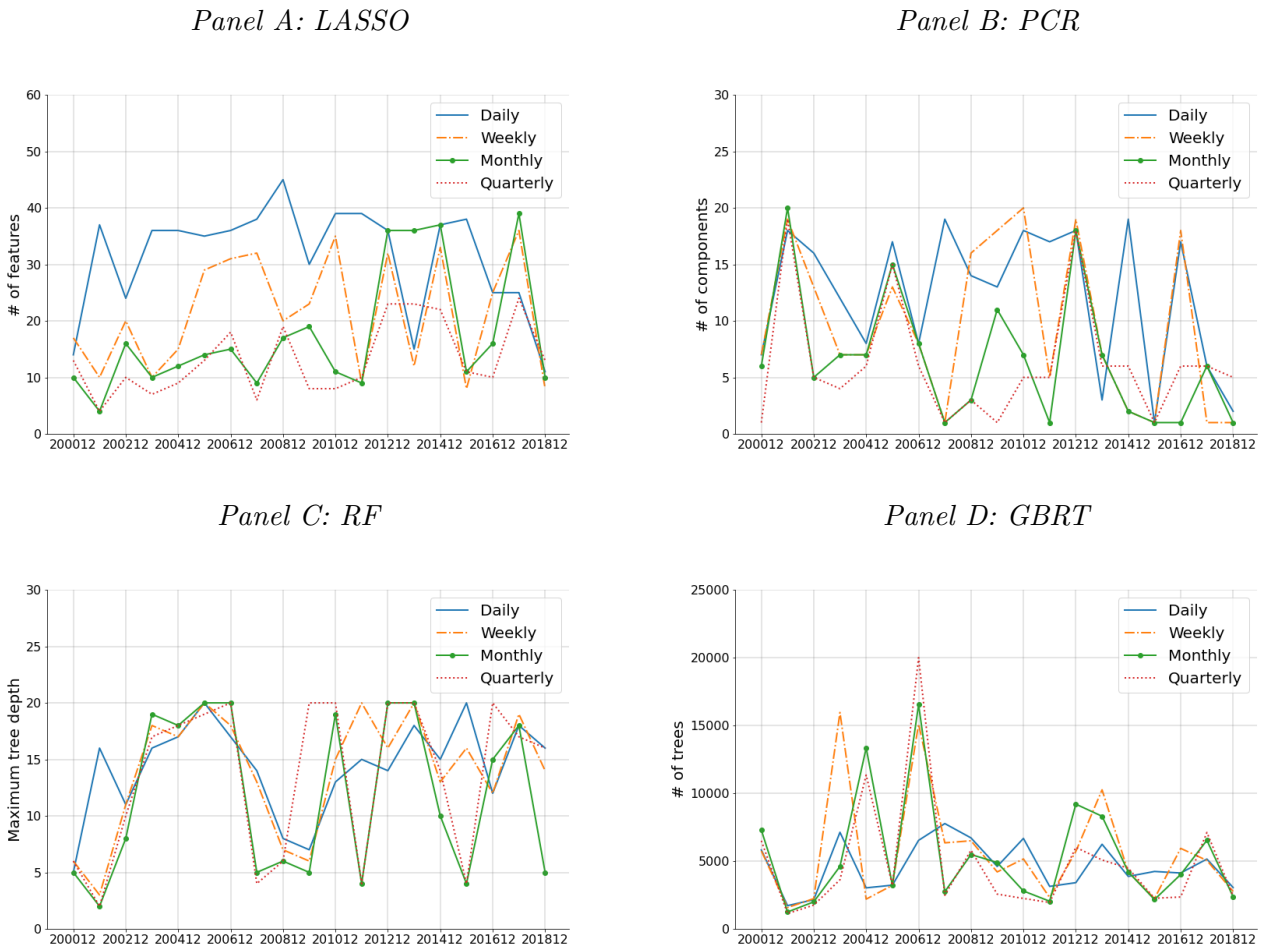



Fig. A.3 Model complexity over time

This figure displays the complexity of LASSO, Principal Component Regression (PCR), Random Forest (RF), and Gradient Boosted Regression Trees (GBRT) validated using each training and validation sample in our out-of-sample analyses across various forecast horizons. Our first training sample is from January 1996 to December 1999 and our first validation sample is from January 2000 to December 2000; our last training sample is from January 1996 to December 2017 and our last validation sample is from January 2018 to December 2018. By the end of each validation sample, we report the number of selected features with nonzero coefficients for LASSO, the number of principal components for PCR, the maximum tree depth for RF, and the total number of trees for GBRT.

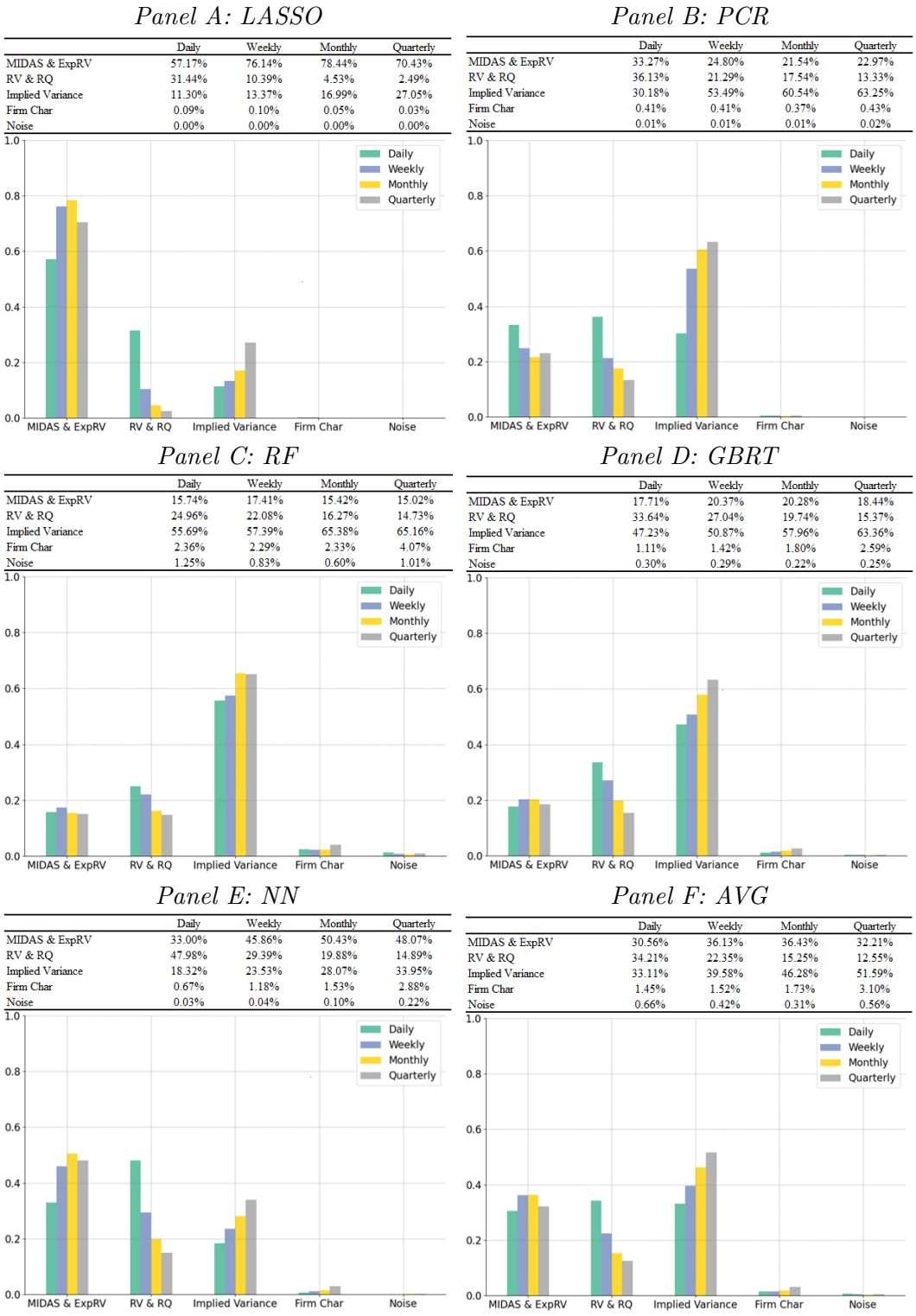



Fig. A.4 Group importance based on 130 Features

This figure displays the group importance based on 130 features for LASSO, PCR, RF, GBRT, NN, and AVG across different forecast horizons. The 130-predictor feature set includes the 118 features used in the main analyses, six firm characteristics, and six noise terms. The first group "MIDAS & ExpRV" includes the $MIDAS$ term for the corresponding forecast horizon, $ExpRV^1$, $ExpRV^5$, $ExpRV^{25}$, $ExpRV^{125}$, and $ExpGlRV$. The second group "RV& RQ" includes $RV^d$, $RV^w$, $RV^m$, $RV^q$, $RVP^d$, $RVN^d$, $RV^d\sqrt{RQ^d}$, $RV^w\sqrt{RQ^w}$, $RV^m\sqrt{RQ^m}$, and $RV^q\sqrt{RQ^q}$. The third group "Implied Variance" includes $CIV^{jm,\delta}$ and $PIV^{jm,-\delta}$, where $j = 1, 2, 3$, and $\delta = 0.1, 0.15, ..., 0.9$. The fourth group "Firm Char" includes firm size, book-to-market ratio, momentum, lagged daily return, lagged monthly return, and illiquidity. The last group "Noise" includes six noise terms generated according to Eq. (A.17).

## Table A.1 Sample Construction and Descriptive statistics

Panel A reports the steps to construct the final stock sample along with the average number of strike prices, call contracts, and put contracts per stock. Panel B reports descriptive statistics for all realized features and selective implied variance features with absolute delta equal to 0.5. The final sample consists of 173 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 100 index between January 1996 and June 2019 with share code 10 or 11, prices between \$1 and \$1000, daily number of trades greater than or equal to 100, and at least five years of data on all features and response variables. Superscripts $d$, $w$, $m$, and $q$ are abbreviations of daily, weekly, monthly, and quarterly construction intervals or forecast horizons. $MIDAS^k$ $(k = d, w, m, q)$ denotes the smoothly weighted moving average of 50 lagged realized variances using validated polynomials from Eqs. (A.2) and (A.3) in forecasting realized variance at horizon $k$. $RV^k$ $(k = d, w, m, q)$ is the daily, weekly, monthly or quarterly realized variance. $RVP^d$ and $RVN^d$ are the daily realized positive and negative semivariances, respectively. $RV^k\sqrt{RQ^k}$ $(k = d, w, m, q)$ is the product of the realized variance and the square root of the realized quarticity with the same construction interval $k$. $ExpRV^i$ $(i = 1, 5, 25, 125)$ is the exponentially weighted moving average of the past 500-day realized variances using the corresponding center-of-mass $i$. $ExpGlRV$ is the exponentially weighted moving average of the global risk factor with a 5-day center-of-mass. $CIV^{jm,0.5}$ and $PIV^{jm,-0.5}$ are implied variances from call and put options with absolute delta equal to 0.5 and maturity equal to $j$ months $(j = 1, 2, 3)$.

Panel A: Sample construction

| | S&P 100 | S&P 500 |
|---|---|---|
| Historical Constituents | 185 | 961 |
| Stocks with Option(s) Traded | 185 (100%) | 961 (100%) |
| Stocks with IV Features | 184 (99%) | 947 (99%) |
| Stocks with RV & IV Features | 173 (94%) | 836 (87%) |
| Avg Number of Unique Strike Prices per Stock | 29 | 21 |
| Avg Number of Call Contracts per Stock | 138 | 89 |
| Avg Number of Put Contracts per Stock | 138 | 89 |

Panel B: Descriptive statistics for realized and selective implied features

| | Mean | Std | Skewness | Kurtosis | P1 | P5 | Median | P95 | P99 | AR(1) | AR(5) | AR(21) | AR(63) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $MIDAS^d$ | 0.145 | 0.243 | 7.575 | 106.664 | 0.012 | 0.019 | 0.076 | 0.478 | 1.143 | 0.969 | 0.839 | 0.629 | 0.457 |
| $MIDAS^w$ | 0.145 | 0.236 | 7.428 | 102.063 | 0.013 | 0.020 | 0.078 | 0.471 | 1.116 | 0.985 | 0.905 | 0.688 | 0.489 |
| $MIDAS^m$ | 0.145 | 0.233 | 7.344 | 99.184 | 0.013 | 0.020 | 0.079 | 0.468 | 1.103 | 0.991 | 0.933 | 0.725 | 0.508 |
| $MIDAS^q$ | 0.145 | 0.228 | 7.217 | 94.686 | 0.014 | 0.021 | 0.081 | 0.464 | 1.089 | 0.995 | 0.960 | 0.780 | 0.534 |
| $RV^d$ | 0.144 | 0.299 | 9.249 | 152.262 | 0.009 | 0.014 | 0.065 | 0.507 | 1.349 | 0.581 | 0.466 | 0.366 | 0.280 |
| $RV^w$ | 0.148 | 0.265 | 7.899 | 115.693 | 0.011 | 0.017 | 0.073 | 0.509 | 1.258 | 0.945 | 0.656 | 0.508 | 0.382 |
| $RV^m$ | 0.150 | 0.247 | 8.076 | 119.655 | 0.014 | 0.021 | 0.081 | 0.487 | 1.135 | 0.993 | 0.945 | 0.682 | 0.482 |
| $RV^q$ | 0.151 | 0.235 | 7.504 | 97.368 | 0.017 | 0.024 | 0.087 | 0.471 | 1.103 | 0.999 | 0.989 | 0.910 | 0.612 |
| $RVP^d$ | 0.072 | 0.158 | 11.268 | 251.878 | 0.004 | 0.006 | 0.031 | 0.255 | 0.684 | 0.513 | 0.414 | 0.324 | 0.248 |
| $RVN^d$ | 0.070 | 0.155 | 10.070 | 189.623 | 0.003 | 0.006 | 0.030 | 0.252 | 0.687 | 0.495 | 0.400 | 0.317 | 0.238 |
| $RV^d\sqrt{RQ^d}$ | 0.257 | 3.111 | 45.794 | 3351.632 | 0.000 | 0.000 | 0.007 | 0.504 | 4.198 | 0.259 | 0.169 | 0.116 | 0.079 |
| $RV^w\sqrt{RQ^w}$ | 0.281 | 2.272 | 25.820 | 1024.559 | 0.000 | 0.001 | 0.012 | 0.733 | 5.394 | 0.853 | 0.281 | 0.180 | 0.116 |
| $RV^m\sqrt{RQ^m}$ | 0.285 | 2.023 | 29.907 | 1495.195 | 0.000 | 0.001 | 0.020 | 0.964 | 4.888 | 0.973 | 0.837 | 0.315 | 0.176 |
| $RV^q\sqrt{RQ^q}$ | 0.287 | 1.820 | 29.610 | 1455.167 | 0.001 | 0.002 | 0.031 | 1.026 | 4.252 | 0.994 | 0.962 | 0.783 | 0.291 |
| $ExpRV^1$ | 0.148 | 0.274 | 8.341 | 128.445 | 0.011 | 0.017 | 0.072 | 0.508 | 1.269 | 0.875 | 0.625 | 0.477 | 0.357 |
| $ExpRV^5$ | 0.149 | 0.254 | 8.064 | 120.020 | 0.013 | 0.020 | 0.079 | 0.496 | 1.168 | 0.976 | 0.863 | 0.626 | 0.445 |
| $ExpRV^{25}$ | 0.151 | 0.235 | 7.447 | 97.015 | 0.017 | 0.024 | 0.087 | 0.476 | 1.093 | 0.997 | 0.978 | 0.869 | 0.624 |
| $ExpRV^{125}$ | 0.154 | 0.200 | 5.232 | 42.188 | 0.021 | 0.029 | 0.097 | 0.464 | 1.017 | 1.000 | 0.997 | 0.978 | 0.890 |
| $ExpGlRV$ | 0.178 | 0.294 | 6.849 | 81.002 | 0.021 | 0.030 | 0.094 | 0.603 | 1.454 | 0.993 | 0.942 | 0.743 | 0.520 |
| $CIV^{1m,0.5}$ | 0.126 | 0.167 | 5.913 | 63.823 | 0.015 | 0.022 | 0.077 | 0.384 | 0.819 | 0.972 | 0.921 | 0.793 | 0.635 |
| $CIV^{2m,0.5}$ | 0.123 | 0.158 | 5.953 | 65.884 | 0.016 | 0.023 | 0.076 | 0.367 | 0.771 | 0.982 | 0.946 | 0.840 | 0.670 |
| $CIV^{3m,0.5}$ | 0.118 | 0.146 | 5.643 | 57.782 | 0.016 | 0.023 | 0.075 | 0.348 | 0.719 | 0.988 | 0.959 | 0.868 | 0.700 |
| $PIV^{1m,-0.5}$ | 0.132 | 0.200 | 11.795 | 305.546 | 0.016 | 0.023 | 0.079 | 0.395 | 0.860 | 0.977 | 0.930 | 0.803 | 0.642 |
| $PIV^{2m,-0.5}$ | 0.129 | 0.191 | 12.798 | 359.667 | 0.018 | 0.025 | 0.080 | 0.377 | 0.807 | 0.985 | 0.953 | 0.852 | 0.681 |
| $PIV^{3m,-0.5}$ | 0.126 | 0.181 | 13.983 | 431.430 | 0.019 | 0.027 | 0.080 | 0.359 | 0.755 | 0.990 | 0.965 | 0.880 | 0.714 |

Table A.2 Out-of-sample prediction relative to long-run mean: OLS-based models

This table reports the out-of-sample $R^2$ relative to the historical mean of realized volatilities for OLS-based volatility forecasting models across different forecast horizons. The sample consists of 173 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 100 index between January 1996 and June 2019 with share code 10 or 11, prices between \$1 and \$1000, daily number of trades greater than or equal to 100, and at least five years of data on all features and response variables. The full out-of-sample evaluation period is from January 2001 to June 2019. Superscripts $d$, $w$, $m$, and $q$ are abbreviations of daily, weekly, monthly, and quarterly construction intervals or forecast horizons. $MIDAS$ denotes the smoothly weighted moving average of 50 lagged realized variances using validated polynomials for the corresponding forecast horizon. $RV^k$ $(k = d, w, m, q)$ is the daily, weekly, monthly or quarterly realized variance. $RVP^d$ and $RVN^d$ are the daily realized positive and negative semivariances, respectively. $RV^k\sqrt{RQ^k}$ $(k = d, w, m, q)$ is the product of the realized variance and the square root of the realized quarticity with the same construction interval $k$. $ExpRV^i$ $(i = 1, 5, 25, 125)$ is the exponentially weighted moving average of the past 500-day realized variances using the corresponding center-of-mass $i$. $ExpGlRV$ is the exponentially weighted moving average of the global risk factor with a 5-day center-of-mass. $CIV^{jm,\delta}$ and $PIV^{jm,-\delta}$ are implied variances from call and put options with absolute $\delta = 0.1, 0.15, ..., 0.9$ and maturity equal to $j$ months $(j = 1, 2, 3)$. Our OLS-based models include MIDAS, SHAR, HARQ-F, HExpGl, $OLS^{RM}$ (i.e., simple OLS model with all 16 realized features as predictors), $OLS^{IV}$ (i.e., simple OLS model with all 102 implied variance features as predictors), and $OLS^{ALL}$ (i.e., simple OLS model with all 118 realized and implied variance features as joint predictors). $R^2_{OOS}$ for each model at each forecast horizon is calculated relative to the long-run mean of $RV$ using the entire panel of stocks according to Eq. (5).

| Model | Features | Daily | Weekly | Monthly | Quarterly |
|---|---|---|---|---|---|
| | | $R^2_{OOS}$ relative to long-run mean | | | |
| HAR | $RV^d$, $RV^w$, $RV^m$, $RV^q$ | 57.8% | 69.4% | 70.0% | 63.6% |
| MIDAS | $MIDAS$ term for the corresponding forecast horizon | 58.2% | 70.6% | 71.3% | 64.2% |
| SHAR | $RVP^d$, $RVN^d$, $RV^w$, $RV^m$, $RV^q$ | 58.4% | 69.9% | 70.4% | 63.9% |
| HARQ-F | $RV^d$, $RV^w$, $RV^m$, $RV^q$, $RV^d\sqrt{RQ^d}$, $RV^w\sqrt{RQ^w}$, $RV^m\sqrt{RQ^m}$, $RV^q\sqrt{RQ^q}$ | 58.7% | 70.3% | 71.0% | 65.4% |
| HExpGl | $ExpRV^1$, $ExpRV^5$, $ExpRV^{25}$, $ExpRV^{125}$, $ExpGlRV$ | 57.8% | 70.2% | 70.6% | 63.1% |
| $OLS^{RM}$ | $MIDAS$ term for the corresponding forecast horizon, $RV^d$, $RV^w$, $RV^m$, $RV^q$, $RVP^d$, $RVN^d$, $RV^d\sqrt{RQ^d}$, $RV^w\sqrt{RQ^w}$, $RV^m\sqrt{RQ^m}$, $RV^q\sqrt{RQ^q}$, $ExpRV^1$, $ExpRV^5$, $ExpRV^{25}$, $ExpRV^{125}$, $ExpGlRV$ (# of features = 16) | 59.8% | 71.4% | 71.6% | 64.3% |
| $OLS^{IV}$ | $CIV^{jm,\delta}$ and $PIV^{jm,-\delta}$, $j = 1, 2, 3$, $\delta = 0.1, 0.15, ..., 0.9$ (# of features = 102) | 53.6% | 67.2% | 69.1% | 62.9% |
| $OLS^{ALL}$ | All 118 Features (16 realized features + 102 $IV$ features) | 61.0% | 73.0% | 72.2% | 63.4% |

Table A.3 Out-of-sample predictions relative to long-run mean: Machine-learning-based models

This table reports the out-of-sample $R^2$ relative to the historical mean of realized volatilities for machine-learning-based volatility forecasting models across different forecast horizons. The sample consists of 173 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 100 index between January 1996 and June 2019 with share code 10 or 11, prices between \$1 and \$1000, daily number of trades greater than or equal to 100, and at least five years of data on all features and response variables. The full out-of-sample evaluation period is from January 2001 to June 2019. The features of each model consist of all 118 predictors detailed in Table 2. Our machine-learning-based models include LASSO, Principal Component Regression (PCR), Random Forest (RF), Gradient Boosted Regression Trees (GBRT), Neural Network (NN), and a simple average of forecasts from the five individual machine learning models (AVG). Tuning parameters for each model are in **bold**. $R^2_{OOS}$ for each model at each forecast horizon is calculated relative to the long-run mean of $RV$ using the entire panel of stocks according to Eq. (5).

| Model | Hyperparameter (Tuning parameter in **bold**) | Daily | Weekly | Monthly | Quarterly |
|---|---|---|---|---|---|
| | | $R^2_{OOS}$ relative to long-run mean | | | |
| LASSO | **# of shrinkage parameters ($\lambda$): 100**<br>$\lambda_{min}/\lambda_{max}$: 0.001 | 61.1% | 73.1% | 73.4% | 64.6% |
| PCR | **# of components: 1, 2, ..., 20** | 60.1% | 70.9% | 72.4% | 66.5% |
| RF | **Maximum tree depth ($L$): 1, 2, ..., 20**<br># of trees: 500<br>Subsample: 0.5<br>Subfeature: ln(# of features) | 59.1% | 71.4% | 72.8% | 65.6% |
| GBRT | **# of trees ($B$)**<br>**Maximum tree depth ($L$): 1, 2, ..., 5**<br>Learning rate: 0.001<br>Subsample: 0.5<br>Subfeature: ln(# of features)<br>Early-stopping rules (whichever met first):<br>1) No reduction in MSE after 50 iterations<br>2) Max # of trees hit 20,000 | 59.8% | 72.6% | 73.2% | 65.9% |
| NN | # of hidden layer: 2<br># of neurons: (5, 2)<br>Activation function: ReLU | 62.2% | 74.5% | 74.3% | 65.4% |
| AVG | | 61.6% | 73.8% | 74.5% | 67.2% |

Table A.4 Out-of-sample prediction relative to HAR: Firm Characteristics and noise terms

This table reports the out-of-sample $R^2$ relative to the HAR model for OLS-based and machine-learning-based volatility forecasting models across different forecast horizons using all 130 predictors, including 118 predictors used in the main analyses, six cross-sectionally ranked firm characteristics, and six pure noise terms. The sample consists of 173 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 100 index between January 1996 and June 2019 with share code 10 or 11, prices between \$1 and \$1000, daily number of trades greater than or equal to 100, and at least five years of data on all features and response variables. The full out-of-sample evaluation period is from January 2001 to June 2019. In Panel A we replicate the results reported in Table 3 for 118 features, and in Panel B we report the results based on 130 features. Firm characteristics include firm size ($Size$), book-to-market ratio ($BM$), momentum ($Mom$), lagged daily return ($Ret^d$), lagged monthly return ($Ret^m$), and illiquidity ($ILLIQ^m$). We cross-sectionally rank each firm characteristic on each day and map these ranks into the [0,1] interval. Our machine-learning-based models include LASSO, Principal Component Regression (PCR), Random Forest (RF), Gradient Boosted Regression Trees (GBRT), Neural Network (NN), and a simple average of forecasts from the five individual machine learning models (AVG). Tuning parameters for each model are reported in **bold**. $R^2_{OOS}$ for each model is calculated relative to the prediction from HAR using the entire panel of stocks according to Eq. (5).

| Model | Hyperparameter | *Panel A: 118 Features* | | | | *Panel B: 130 Features* | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Daily | Weekly | Monthly | Quarterly | Daily | Weekly | Monthly | Quarterly |
| | | $R^2_{OOS}$ Relative to HAR | | | | | | | |
| $OLS^{ALL}$ | | 7.6% | 11.6% | 7.3% | -0.6% | 7.6% | 11.5% | 6.9% | -1.3% |
| LASSO | **# of shrinkage parameters ($\lambda$): 100**<br>$lambda_{min}/\lambda_{max}$: 0.001 | 8.0% | 12.1% | 11.3% | 2.6% | 7.9% | 12.1% | 11.3% | 2.6% |
| PCR | **# of components: 1,2,. . . ,20** | 5.5% | 4.8% | 8.1% | 7.8% | 5.2% | 3.1% | 8.2% | 7.8% |
| RF | **Maximum tree depth: 1,2,. . . ,20**<br># of trees: 500<br>subsample: 0.5<br>subfeature: ln(# of features) | 3.2% | 6.4% | 9.5% | 5.4% | 2.8% | 6.7% | 10.1% | 5.9% |
| GBRT | **# of trees ($B$)**<br>**Maximum tree depth ($L$): 1, 2, ..., 5**<br>Learning rate: 0.001<br>Subsample: 0.5<br>Subfeature: ln(# of features)<br>Early-stopping rules (whichever met first):<br>1) No reduction in MSE after 50 iterations<br>2) Max # of trees hits 20,000 | 4.7% | 10.2% | 10.8% | 6.3% | 4.7% | 10.7% | 11.0% | 6.5% |
| NN | # of hidden layer: 2<br># of neurons: (5, 2)<br>activation function: ReLU | 10.5% | 16.7% | 14.3% | 4.8% | 10.5% | 16.2% | 12.0% | 1.2% |
| AVG | | 9.0% | 14.3% | 15.2% | 10.0% | 9.1% | 14.5% | 15.2% | 9.7% |

Table A.5 $R^2_{OOS}$ from AVG sorted by firm characteristics

This table reports the time-series mean of within-month average $R^2_{OOS}$ from AVG sorted by firm characteristics across different forecast horizons, where $R^2_{OOS}$ is the out-of-sample $R^2$ relative to the HAR model. The sample consists of 173 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 100 index between January 1996 and June 2019 with share code 10 or 11, prices between \$1 and \$1000, daily number of trades greater than or equal to 100, and at least five years of data on all features and response variables. The full out-of-sample evaluation period is from January 2001 to June 2019. AVG is the simple average of forecasts from the five individual machine learning models based on 118 predictors. Firm characteristics include firm size ($Size$), book-to-market ratio ($BM$), momentum ($Mom$), lagged daily return ($Ret^d$), lagged monthly return ($Ret^m$), and illiquidity ($ILLIQ$). By the end of each month, we sort all stocks into quintile portfolios by a given firm characteristic and compute for each portfolio the $R^2_{OOS}$ relative to HAR based on model AVG in the following month. Then we report the time-series mean of the $R^2_{OOS}$'s. The column labeled "HML" reports the difference in average $R^2_{OOS}$ between portfolio 5 and portfolio 1, with Newey-West robust $t$-statistics in parentheses.

*Panel A: Daily*

| | 1 (Low) | 2 | 3 | 4 | 5 (High) | HML |
|---|---|---|---|---|---|---|
| $Size$ | 7.2% | 9.0% | 9.4% | 10.7% | 12.3% | 5.1% |
| | | | | | | (6.60) |
| $BM$ | 9.9% | 10.2% | 9.2% | 8.2% | 6.5% | -3.4% |
| | | | | | | (-4.52) |
| $Mom$ | 8.3% | 8.6% | 9.8% | 9.6% | 9.2% | 0.9% |
| | | | | | | (1.24) |
| $Ret^d$ | 8.6% | 8.9% | 9.2% | 8.5% | 9.1% | 0.5% |
| | | | | | | (0.71) |
| $Ret^m$ | 9.2% | 8.4% | 8.8% | 9.7% | 8.3% | -0.9% |
| | | | | | | (-1.23) |
| $ILLQ$ | 11.1% | 9.8% | 9.7% | 9.9% | 7.3% | -3.9% |
| | | | | | | (-5.06) |

*Panel B: Weekly*

| | 1 (Low) | 2 | 3 | 4 | 5 (High) | HML |
|---|---|---|---|---|---|---|
| $Size$ | 14.8% | 19.5% | 19.2% | 21.3% | 23.0% | 8.2% |
| | | | | | | (5.27) |
| $BM$ | 19.7% | 21.0% | 18.5% | 15.9% | 12.9% | -6.8% |
| | | | | | | (-4.39) |
| $Mom$ | 16.5% | 17.7% | 19.9% | 19.0% | 19.2% | 2.8% |
| | | | | | | (2.14) |
| $Ret^d$ | 16.3% | 17.6% | 18.4% | 18.2% | 18.5% | 2.2% |
| | | | | | | (1.48) |
| $Ret^m$ | 18.6% | 17.5% | 17.3% | 18.4% | 18.1% | -0.5% |
| | | | | | | (-0.36) |
| $ILLQ$ | 20.0% | 21.2% | 18.8% | 20.5% | 14.9% | -5.1% |
| | | | | | | (-3.24) |

*Panel C: Monthly*

| | 1 (Low) | 2 | 3 | 4 | 5 (High) | HML |
|---|---|---|---|---|---|---|
| $Size$ | 19.8% | 25.9% | 26.1% | 28.2% | 28.9% | 9.1% |
| | | | | | | (3.50) |
| $BM$ | 28.6% | 28.0% | 25.2% | 21.0% | 16.9% | -11.7% |
| | | | | | | (-4.47) |
| $Mom$ | 21.3% | 27.1% | 26.7% | 23.5% | 23.8% | 2.6% |
| | | | | | | (1.16) |
| $Ret^d$ | 23.1% | 24.9% | 26.3% | 24.9% | 22.0% | -1.1% |
| | | | | | | (-0.48) |
| $Ret^m$ | 23.4% | 25.0% | 24.5% | 25.3% | 23.3% | -0.1% |
| | | | | | | (-0.03) |
| $ILLQ$ | 26.0% | 28.4% | 26.3% | 25.3% | 19.3% | -6.7% |
| | | | | | | (-2.62) |

*Panel D: Quarterly*

| | 1 (Low) | 2 | 3 | 4 | 5 (High) | HML |
|---|---|---|---|---|---|---|
| $Size$ | 15.7% | 19.7% | 21.4% | 24.8% | 26.0% | 10.3% |
| | | | | | | (3.23) |
| $BM$ | 26.3% | 26.5% | 23.2% | 18.2% | 14.2% | -12.1% |
| | | | | | | (-4.22) |
| $Mom$ | 17.8% | 23.8% | 25.9% | 21.4% | 15.1% | -2.7% |
| | | | | | | (-1.04) |
| $Ret^d$ | 20.6% | 20.9% | 23.6% | 23.9% | 17.1% | -3.4% |
| | | | | | | (-1.33) |
| $Ret^m$ | 21.3% | 21.7% | 23.2% | 22.9% | 16.6% | -4.7% |
| | | | | | | (-1.63) |
| $ILLQ$ | 22.2% | 23.8% | 23.0% | 20.3% | 14.2% | -8.1% |
| | | | | | | (-2.76) |

## Table A.6 Out-of-sample prediction relative to HAR: Robustness

This table reports the out-of-sample $R^2$ relative to the HAR model for OLS-based and machine-learning-based volatility forecasting models across different forecast horizons in a robustness analysis. The sample consists of stocks listed on NYSE/AMEX/NASDAQ with share code 10 or 11, prices between \$1 and \$1000, and daily number of trades greater than or equal to 100. The full out-of-sample evaluation period is from January 2001 to June 2019. Panel A focuses on a sample consisting of 205 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 100 index between January 1996 and June 2019, relaxing the five-year data requirement. Panel B evaluates the out-of-sample performance for a sample consisting of 836 stocks listed on NYSE/AMEX/NASDAQ that have ever been included in the S&P 500 index between January 1996 and June 2019, excluding forecasts for periods when a stock has not entered into the index. The features of each OLS-based model consist of either model-specific predictors or all 118 predictors as detailed in Table 2, and those of each ML-based model consist of all 118 predictors. Our ML-base models include LASSO, Principal Component Regression (PCR), Random Forest (RF), Gradient Boosted Regression Trees (GBRT), Neural Network (NN), and a simple average of forecasts from the five individual machine learning models (AVG). $R^2_{OOS}$ for each model is calculated relative to the prediction from HAR using the entire panel of stocks according to Eq. (5).

| | | *Panel A: Relax five-year requirement* | | | | *Panel B: Exclude before inclusion* | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Daily | Weekly | Monthly | Quarterly | Daily | Weekly | Monthly | Quarterly |
| | | $R^2_{OOS}$ *Relative to HAR* | | | | $R^2_{OOS}$ *Relative to HAR* | | | |
| OLS | MIDAS | 1.1% | 3.7% | 4.3% | 1.6% | 0.6% | 2.1% | 1.9% | -0.1% |
| | SHAR | 1.5% | 1.6% | 1.4% | 0.7% | 1.0% | 1.2% | 0.9% | 0.5% |
| | HARQ-F | 2.2% | 3.0% | 3.9% | 5.4% | 1.0% | 1.3% | 0.7% | 1.0% |
| | HExpGl | 0.0% | 2.4% | 2.1% | -1.3% | 0.3% | 2.0% | 1.9% | -0.4% |
| | $OLS^{RM}$ | 4.9% | 6.6% | 6.3% | 3.5% | 3.5% | 5.0% | 3.1% | 0.1% |
| | $OLS^{IV}$ | -10.4% | -8.5% | -3.9% | -3.0% | -16.4% | -19.7% | -16.0% | -10.5% |
| | $OLS^{ALL}$ | 7.6% | 11.5% | 7.9% | 1.0% | 5.2% | 8.6% | 5.6% | -0.3% |
| ML | LASSO | 7.8% | 11.9% | 11.4% | 4.5% | 5.5% | 9.3% | 8.2% | 2.5% |
| | PCR | 6.1% | 4.2% | 7.4% | 8.6% | 4.3% | 6.4% | 4.8% | 6.3% |
| | RF | 3.5% | 6.7% | 11.0% | 7.6% | 4.8% | 7.9% | 7.2% | 4.6% |
| | GBRT | 4.7% | 10.1% | 11.2% | 7.6% | 5.8% | 10.5% | 7.6% | 3.5% |
| | NN | 10.7% | 16.8% | 14.2% | 3.0% | 8.9% | 14.9% | 11.7% | 4.2% |
| | AVG | 9.1% | 14.4% | 15.5% | 10.8% | 7.7% | 13.0% | 11.6% | 7.5% |