Erasmus University Rotterdam
Erasmus School of Economics
Bachelor Econometrics and Operations research

Combining HAR models and a feed-forward neural network to forecast one day-ahead realized variance

Ole Dorhout Mees (585046)

Supervisor:	dr. Onno Kleen
Second assessor:	Max F.O. Welz
Date:	July 2, 2023

Abstract

In this research paper, we examine the impact of combining heterogeneous autoregressive (HAR) models with a feed-forward neural network on one day-ahead realized variance (RV) forecasting performance. The study utilizes volatility data from 25 stocks in the Dow Jones Industrial Average Index spanning January 2001 to December 2021. We use six variations of the HAR model, a feed-forward neural network, and three model combinations. We assess forecasts using the mean squared error, QLIKE, and Diebold-Mariano tests. Findings show that the simple model combination significantly outperforms the benchmark HAR, semivariance HAR, and log HAR model, but does not significantly improve on other models. Moreover, the simple model combination slightly outperforms the more complex combinations. Overall, the model combination provides robust RV forecasts due to stable performance in both low and high volatility regimes. This research emphasizes the importance of combining forecasting models to achieve robust RV forecasts while maintaining computational efficiency.

Keywords: Realized variance forecasting, Heterogeneous autoregressive (HAR) model, Machine Learning, Feed-forward neural network, Model combinations, peLASSO.

1 Introduction

Over the past decades, volatility forecasting has become increasingly important due to the increase in market volatility. With the availability of high-frequency data and advancements in machine learning (ML), there has been a shift in the focus of volatility forecasting research. Realized variance (RV) has emerged as the standard measure of volatility, and ML techniques, such as neural networks, have shown promising results in short-horizon forecasts, while tree-based models excel in long-horizon forecasts (Christensen et al., 2022).

However, not every researcher or business has access to large computational capacities in order to run computationally demanding ML algorithms. Additionally, hyperparameter tuning of ML models can be challenging and opaque. Therefore, there is a need for clear and computationally efficient methods that can be readily used by researchers and analysts facing these limitations.

To address these challenges, this research explores the combination of traditional, easy-to-use heterogeneous autoregressive (HAR) models with an off-the-shelf feed-forward neural network (FFNN) to forecast one-day-ahead RV. The aim is to assess the extent to which a model combination of HAR models and a neural network can improve volatility forecasting performance. Furthermore, the research investigates the performance of these models and combinations in different volatility regimes and compares the forecasting performance of simple and complex model combinations.

To answer our research questions, we use data from various sources. We obtain our data from Kleen and Tetereva (2022), OptionMetrics, and Yahoo Finance. We use seven explanatory variables on 25 companies with observations ranging from January 29, 2001 to December 30, 2021 (5264 observations). The dataset is split into a training, validation, combination, and test set. The validation set is used to optimize the FFNN and prevent overfitting. The combination set is used to estimate forecast weights of the model combinations. The dataset used by Kleen and Tetereva (2022) includes: RV, semivariance RV (RV-/RV), quadratic variation (QV), and the CBOE volatility index (VIX). The daily (RVD), weekly (RVM), and monthly (RVM) lags of RV are computed from the provided RV. From OptionMetrics by WRDS, we obtain the implied volatility (IV). This variable captures the expected volatility which is calculated using option pricing (Jiang & Tian, 2005). Literature finds that the IV is an important predictor of RV (Christensen et al., 2022; Kambouroudis et al., 2021). From Yahoo Finance, we obtain the 1-week momentum (MOM) and dollar trading volume (TV). A description and the descriptive statistics of the variables are included in Appendix A.1 table 2 and 3.

In this study, we fit seven individual models on the training and validation set, including variations of the HAR model and an FFNN. The HAR models use a rolling window approach, while the FFNN uses a fixed window. Moreover, we construct three model combinations: the equally-weighted model combination (SCOMB), the partially egalitarian LASSO model combination (peLCOMB), and the volatility quantile-base model combination (VQCOMB). The weights for the peLCOMB and VQCOMB are estimated on the combination set. Performance comparisons are conducted on the combination and test set, evaluating RV forecasts using metrics such as QLIKE, relative mean squared error (MSE), and the Diebold-Mariano (DM) test.

Previous literature suggests that future volatility can be mostly explained by historical volatility (Christensen et al., 2022). Therefore, the HAR model family is a great method to forecast RV as it uses previous volatility. Extensions on the HAR have been made to leverage various patterns in asset volatility. The logHAR is proposed to enable a more efficient estimation using ordinary least squares, as the estimation errors of the log transformed HAR model show more symmetry (Corsi, 2008). The HAR quarticity (HARQ) and HARQ-full (HARQF), proposed by Bollerslev et al. (2016), incorporate a measure for the measurement error. Subsequently, the models allow for greater persistence of previous volatility when measurement errors are small, and lesser persistence when measurement errors are large. The semivariance HAR (SHAR) is suggested to capture the 'leverage effect' using RV- and RV+ (Patton & Sheppard, 2015). However, Christensen et al. (2022) also indicates that expected volatility is an important predictor of RV. The HARX model incorporates other lagged variables in addition to the nor-

mal HAR structure. IV allows the HARX model to include expected volatility, which has been proven to be of significant importance (Kambouroudis et al., 2021).

As previously mentioned, ML is a well-performing model class in volatility forecasting and mainly includes tree-based models and neural networks. Tree-based models show favorable performance, particularly for longer forecasting horizons, as highlighted by Christensen et al. (2022). However, since we focus on one day-ahead forecasting, we decide to research the neural networks as these show more promising results in shorter horizons. Current literature on neural networks in forecasting RV predominantly focuses on the long short-term recurrent neural network (LSTM RNN) as this shows better results (Bucci, 2020; Rahimikia and Poon, 2020; Zhang et al., 2023). However, the LSTM RNN is computationally demanding and therefore not suited for this research. Therefore, we decide to implement the FFNN as previously research by Christensen et al. (2022).

Christensen et al. (2022) observe varying performance between HAR models and ML models across various volatility regimes. HAR models demonstrate relatively favorable performance in periods of low volatility, whereas FFNN shows promising results on mid and high volatility. Thus, the combination of HAR forecasts with FFNN forecasts may prove beneficial. Forecast combinations are well-established and effective approaches for generating robust forecasts. Extensive insights into forecast combinations are provided by Timmermann (2006), who highlights the effectiveness of simple combinations over complex ones. Timmermann (2006) finds that allowing for modest time variations in weights and eliminating poorly performing models, prove to be useful. The peLASSO method, introduced by Diebold and Shin (2019), uses a shrinkage method to eliminate underperforming models while assigning equal weight to the remaining models. This approach promotes simplicity in the model structure while deleting unnecessary models. In addition, this research proposes the VQCOMB to allow for varying model weights over various volatility regimes.

The results reveal that the SCOMB model combination showed significant improvements over the benchmark HAR model, as well as the SHAR and logHAR models showing significant reductions in MSE compared to the HAR model (13.3%, 7.8%, and 35.6% respectively). However, the SCOMB does not outperform the HARQ(F), HARX, and FFNN models. In low volatility regimes, the logHAR model achieves a significant 65% MSE reduction, while the FFNN model shows a substantial increase of 73% in relative MSE. In high volatility regimes, the HARQF and HARX models have MSE reductions (14% and 21% respectively), whereas the logHAR and FFNN models underperform. The SCOMB model beats the peLCOMB and VQCOMB in terms of MSE across all volatility regimes, highlighting the importance of a simpler model combination

approach. In conclusion, the SCOMB model proves to be a robust choice for RV forecasting, given the varying model performance across different volatility regimes.

This research shows that a model combination is not able to beat all HAR models and the neural network, but allows for more robust forecasts of RV. Especially in low and high volatility, not all forecasting models are capable of providing reasonable RV forecasts, whereas, the SCOMB can provide robust forecasts. Therefore, combining forecast results can increase robustness in these situations. Moreover, this study describes the behavior and performance of the various models in different volatility regimes. Additionally, we identify that simple model combinations are more useful than complex model combinations.

The remainder of this paper is structured as follows. First, we explain how we handle the data in Section 2. Then, we elaborate on the used methods in Section 3. Following the methodology, we show the results in Section 4. Lastly, we present our main findings in Section 5.

2 Data

This section provides a description of the dataset used in our research. The dataset is obtained from multiple sources and consists of daily observations spanning from January 29, 2001, to December 30, 2021, containing 5264 trading days. The dataset contains information related to 25 companies included in the Dow Jones Industrial Average Index, which are labeled using a tracker. The variables included in the dataset are realized variance (RV), realized quarticity (RQ), realized semivariances (RV-/RV+), the CBOE volatility index (VIX), implied volatility (IV), 1-week momentum (MOM), and dollar trading volume (TV). An overview of the variables is provided in Appendix A.1.

Kleen and Tetereva (2022) provide us with a dataset containing: RV, RQ, RV-/RV+, and VIX. This dataset covers 29 companies from January 3, 2000 to December 30, 2021. We extend this dataset by adding IV through OptionMetrics and MOM and TV through Yahoo Finance. The IV data is obtained from OptionMetrics through the WRDS database. We query the implied volatility of options for the 25 companies from January 29, 2001, to December 30, 2021, and calculate the median implied volatility for each asset on a given date. MOM and TV are obtained from daily data sourced from Yahoo Finance, including closing prices and trading volume.

The dataset spans a period that includes significant market events, such as: the dot-com bubble and burst (2000-2001), 9/11 (2001), the global financial crisis (2007-2009), the European debt crisis (2010-2012), the flash crashes of May 6, 2010, and August 24, 2015, the US debt ceiling crisis (2011), and the COVID recession (2020-). We focus on 25 companies, excluding DOW (Dow Chemicals), JPM (JP Morgan), PG (Procter and Gamble), and UNH (United

Health) due to missing observations. The tickers of the included companies are: AAPL, AXP, BA, CAT, CSCO, CVX, DIS, GE, GS, HD, IBM, INTC, JNJ, KO, MCD, MMM, MRK, MSFT, NKE, PFE, RTX, TRV, VZ, WMT, XOM.

To fit our research, we segment the data into different subsets. For replication purposes, we use the same training, validation and testing set as Christensen et al. (2022) but we label the testing set as our combination set. The training set covers the period from January 29, 2001, to December 12, 2012, comprising 57% of the dataset. This used to fit all the individual models. The validation set spans from December 13, 2012, to August 20, 2014, covering 8%. This subsample is used to construct the FFNN. For the HAR models, we incorporate the validation set into the training set. The combination set is defined from August 21, 2014, to December 29, 2017, representing 16% of the data. This subsample is used to estimate combination weights for our model combinations. Finally, the testing set includes observations from December 30, 2017, to December 30, 2021, accounting for 19%.

To forecast realized variance, we rely on the explanatory ability of multiple variables. Christensen et al. (2022) find - according to their variable importance measure - that daily, weekly and monthly lags of the RV, VIX, and the IV are the most important predictors of RV. Therefore, we decide to include these variables in our dataset. Kambouroudis et al. (2021) find that implementing the IV in the HAR is more accurate than any HAR model excluding it. Moreover, we use RQ and RV+/RV-, since well-known variations of the HAR model rely on it. Lastly, we use the MOM of asset prices and the TV of the underlying asset.

We preprocess the data to handle missing values and prepare it for neural network input. (i). We use linear interpolation to interpolate the VIX index on May 4, 2006. (ii). We exclude data on February 2, 2011, from our dataset, since this data was also missing from the original dataset provided by Kleen and Tetereva (2022). (iii). We normalize data using the sample mean and sample variance from the training set to standardize the input for the FFNN.

3 Methodology

The aim of this research is to forecast the quadratic variation (QV). We have $X_i = \{X_{i,1}, ..., X_{i,5264}\}$ which denotes the series of log prices for each day for stock i. The QV of X_i captures the stochastic volatility process and the aggregate jumps of X_i (Christensen et al., 2022). Since the QV is a theoretical variable, we estimate the QV using the realized variance (RV). RV is shown to be a consistent estimator of QV (Barndorff-Nielsen & Shephard, 2002). We denote the log price X_i at day t as $X_{i,t}$. In the Equation below, we show the calculation of the realized variance on day t for stock i:

$$RV_{i,t} = \sum_{j=1}^{n} (r_{i,j})^2 \tag{1}$$

where $r_{i,t}$ denotes the intraday returns of stock i at time j and n is the number of intraday returns.

3.1 HAR model

The heterogeneous autoregressive (HAR) model of Corsi (2008) is widely considered the benchmark in volatility forecasting research (Christensen et al., 2022). To ensure comparability with available research, we use the HAR model as the benchmark model in this research. The HAR model is defined as

$$RV_{i,t} = \beta_{i,0} + \beta_{i,1} RV_{i,t-1} + \beta_{i,2} RV_{i,t-2|t-5} + \beta_{i,3} RV_{i,t-6|t-22} + u_{i,t}, \tag{2}$$

where $u_{i,t} \overset{\text{iid}}{\sim} N(0,1)$. $RV_{i,t-1}$ denotes the lagged daily RV. $RV_{i,t-2|t-5}$ and $RV_{i,t-6|t-22}$ the lagged weekly and monthly RV respectively. We compute the daily, weekly, and monthly RV such that they are independent on each other. To show this, the weekly RV uses the RV from 5 days ahead till two days ahead. The RV from one day-ahead is not used, since this would give collinearity issues with the daily RV. The weekly and monthly RV for stock i at day t are calculated as follows:

$$RV_{i,t-j|t-k} = \frac{1}{k-j+1} \sum_{i=k}^{j} RV_{t-i} \tag{3}$$

where $j = 2$ and $k = 5$ for the weekly RV and $j = 6$ and $k = 22$ for the monthly RV.

3.2 Variations on the HAR model

Since Corsi (2008) proposed the general HAR framework, many other variations on this model have been proposed. In the original paper, Corsi (2008) also proposes the log HAR. The log HAR log transforms all the variables of the HAR model. Since the distribution of the RV is left-skewed and right-fat-tailed, the log transformation of RV values makes the distribution more symmetric, and thus more normally distributed. Consequently, the residual of a log HAR resembles the normal distribution better. This makes the parameter estimation using OLS more efficient. The log HAR for stock i is estimated as follows:

$$\log(RV_{i,t}) = \beta_{i,0} + \beta_{i,1} \log(RV_{i,t-1}) + \beta_{i,2} \log(RV_{i,t-2|t-5}) + \beta_{i,3} \log(RV_{i,t-6|t-22}) + u_{i,t}, \tag{4}$$

where $u_{i,t} \overset{\text{iid}}{\sim} N(0,1)$.

To make RV forecasts, we need to transform the log RV back to RV. Since the residual of the log HAR - $u_{i,t}$ - is standard normally distributed, the expected value of its exponential is equal to $\exp(\frac{\sigma^2}{2})$. Thus, the transformation is as follows:

$$RV_{i,t} = \exp(\beta_{i,0} + \beta_{i,1}\log(RV_{i,t-1}) + \beta_{i,2}\log(RV_{i,t-2|t-5}) + \beta_{i,3}\log(RV_{i,t-6|t-22})) + \exp(\frac{\hat{\sigma^2}}{2}), \quad (5)$$

where $\hat{\sigma^2}$ will be estimated using the sample variance s^2.

Another variation on the HAR model is the semivariance HAR model (SHAR) (Patton & Sheppard, 2015). This model is set up such that it is able to capture the leverage effect. The leverage effect refers to the observed tendency of an asset's volatility to be negatively correlated with the asset's returns (Aït-Sahalia et al., 2013). In other words, volatility tends to increase when the asset price declines. Whereas when there is an increase in price, asset price levels are more stable. The SHAR model leverages these effects by using the positive and negative semivariance. The SHAR model is as follows:

$$RV_{i,t} = \beta_{i,0} + \beta^{-}_{i,1}RV^{-}_{i,t-1} + \beta^{+}_{i,1}RV^{+}_{i,t-1} + \beta_{i,2}RV_{i,t-2|t-5} + \beta_{i,3}RV_{i,t-6|t-22} + u_{i,t}, \quad (6)$$

where $RV^{+}_{i,t} = \sum_{j:r_{i,j}>0}(r_{i,j})^2$, $RV^{-}_{i,t} = \sum_{j:r_{i,j}<0}(r_{i,j})^2$, $r_{i,j}$ is the intraday return of stock i, and $u_{i,t} \overset{\text{iid}}{\sim} N(0,1)$.

Bollerslev et al. (2016) propose the HAR Quarticity (HARQ) and HARQ full (HARQF)to allow for time-varying parameters of the HAR model that differ with the estimated degree of measurement error. The result is that the model can allow for greater persistence when the measurement error is small, and weaker persistence when the measurement error is large. The HARQ and HARQF model use the realized quarticity (RQ): $RQ_{i,t} = \frac{n}{3}\sum_{j=1}^{n}(r_{i,})^4$. RQ is used to estimate the degree of measurement error. RQ has been shown to estimate the integrated quarticity (IQ) (Barndorff-Nielsen & Shephard, 2002), which in turn captures the estimation error of RV. Therefore, RQ is a consistent estimator of the estimation error of RV. The HARQ as proposed by Bollerslev et al. (2016):

$$RV_{i,t} = \beta_{i,0} + (\beta_{i,1} + \beta_{i,1Q}RQ^{1/2}_{i,t-1})RV_{i,t-1} + \beta_{i,2}RV_{i,t-1|t-5} + \beta_{i,3}RV_{i,t-1|t-22} + u_{i,t}, \quad (7)$$

where $u_{i,t} \overset{\text{iid}}{\sim} N(0,1)$.

To allow for longer-term leverage effects, Bollerslev et al. (2016) also propose the HARQF:

$$RV_{i,t} = \beta_{i,0} + (\beta_{i,1} + \beta_{i,1Q} RQ_{i,t-1}^{1/2}) RV_{i,t-1} + (\beta_{i,2} + \beta_{i,2Q} RQ_{i,t-2|t-5}^{1/2}) RV_{i,t-2|t-5} + (\beta_{i,3} + \beta_{i,3Q} RQ_{i,t-6|t-22}^{1/2}) RV_{i,t-6|t-22} + u_{i,t}, \quad (8)$$

where $u_{i,t} \overset{\text{iid}}{\sim} N(0,1)$.

Lastly, we define the HARX model. This model includes the "baseline" HAR model and extends it using other explanatory variables. Besides the daily, weekly, and monthly lagged RV, we use the VIX, implied volatility (IV), 1-week momentum (MOM), and dollar trading volume (TV) as described in Section 2:

$$RV_{i,t} = \beta_{i,0} + \beta_{i,1} RV_{i,t-1} + \beta_{i,2} RV_{i,t-2|t-5} + \beta_{i,3} RV_{i,t-6|t-22} + \beta_{i,4} \text{VIX}_{i,t-1} + \beta_{i,5} \text{IV} i, t-1 + \beta_{i,6} \text{MOM}_{i,t-1} + \beta_{i,7} \text{TV}_{i,t-1} + u_{i,t}, \quad (9)$$

where $u_{i,t} \overset{\text{iid}}{\sim} N(0,1)$.

All HAR models and variations on the HAR models are estimated using a rolling window approach. To limit computational demand, we estimate and re-estimate the model three times. For our forecasts on the combination set, we estimate the models on August 21, 2014, October 2, 2015, and November 14, 2016. For the forecast on the test set, we estimate the model on December 30, 2017, May 2, 2019, and August 31, 2020. For both sets, we fix the backward-looking estimation window at $w = 2986$.

To estimate parameters for all our HAR models, we use ordinary least squares (OLS) estimation. We perform all the estimations and forecasts in the R programming language.

3.3 Feed-forward neural network

The feed-forward neural network (FFNN) is a - relatively - lightweight trainable neural network that can be used for forecasting volatility (Christensen et al., 2022). Compared to the widely-recommended long short-term memory recurrent neural network (LSTM RNN) (Bucci, 2020), the FFNN is less computationally demanding. Therefore, we decide to use the FFNN.

3.3.1 Network architecture

A FFNN consists of an input layer, hidden layer(s), and an output layer. The input layer receives the input data. In our case, these are the explanatory variables such as IV and lagged monthly

RV (RVM). The hidden layers consist of neurons that are able to make nonlinear transformations on the data. Eventually, they will - ideally - learn the relations between variables. Lastly, the output layer provides our output data. In this research, the output layers are comprised of a single neuron that gives a value for the (predicted) realized variance.

A hidden layer l is made up of multiple neurons. We build our FFNN such that there are three hidden layers that contain 64, 32, and 16 neurons. A neuron j receives input data and weighs this according to a weight matrix $\theta^{(l)}$. The input data is weighted and a bias term $b^{(l)}$ is added. Next, the transformed data passes through the activation function. The activation function transforms the data nonlinearly. In this way, an FFNN can capture nonlinearity. Our FFNN uses the Rectified Linear Unit (ReLU) as an activation function.

Thus, we can denote the model prediction as a function of the weighted sums, bias terms, and activation functions:

$$a_t^{\theta_{l+1},b_{l+1}} = g_l\left(\sum_{j=1}^{N_l} \theta_j^{(l)} a_t^{\theta_l,b_l} + b^{(l)}\right), \quad 1 \leq l \leq L, \tag{10}$$

where $a_t^{\theta_{l+1},b_{l+1}}$ is the output from layer l with the input for layer $l+1$, g_l is the activation function, $\theta_j^{(l)}$ is the weighted sum of input $a_t^{\theta_l,b_l}$ in neuron j, $b^{(l)}$ is the bias term of layer l, L is the number of layers, and N_l is the number of neurons in the layers.

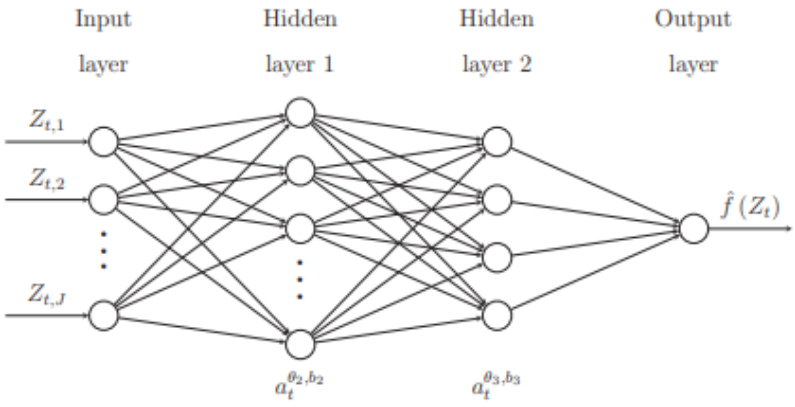


Figure 1: This figure illustrates the architecture of the feed-forward neural network used in this research. This illustration is taken from Christensen et al. (2022).

3.3.2 Parameter estimation

To estimate the parameters, we use back-propagation (Goodfellow et al., 2016). Back-propagation is the practice of using an algorithm to compute the gradient of the loss function (MSE) for the network parameters, the weights $\theta^{(l)}$ and biases $b^{(l)}$ (Goodfellow et al., 2016). The weights

and biases are then updated using the gradient and a certain learning rate. Eventually, the parameters converge to their optimal state. In our case, we use the Adaptive Moment Estimation (Adam) algorithm, since this algorithm allows for fast convergence (Wang et al., 2022).

To better understand the parameter estimation, we will describe the process more closely. First, we select the training and validation set. Second, we initialize the FFNN and compute a prediction. The process of making a prediction using a feed-forward neural network is called forward propagation. Third, we use the Adam algorithm to optimize our parameters according to the loss function (back-propagation). When the model is not 'optimal' yet, we compute another prediction with the 'new' parameters. The process of performing an iteration of forward propagation and back-propagation is called an epoch. Our FFNN algorithm is set up such that we have a maximum of 100 epochs or a loss function that has not changed substantially for 10 epochs. In the latter case, we stop the algorithm at an 'early stage'. In both cases, we complete the parameter estimation.

3.3.3 Regularization

Since the FFNN optimizes its parameters by performing a lot of iterations over the same sample, the model is prone to overfitting. Therefore, to allow for generalization of the model, we use multiple regularization techniques. These include (i). drop out, (ii). early stopping, and (iii). ensembles. Dropout is a technique where a fraction of the hidden layer output is set to zero. Regarding Equation 10, the drop-out function sets values from $a_t^{\theta_{l+1},b_{l+1}}$ to zero. In our FFNN, we employ a drop-out of 0.2. This means that the drop-out function sets 20% of the variables to zero at random. The drop-out function is applied between each hidden layer. Moreover, Early stopping is used to ensure that the FFNN does not perform the maximum amount of epochs each time. This ensures better calculation times, but also regularization. Lastly, ensembling is the technique of combining the models to make predictions. We estimate 100 models of which we select the best ten to be used for the ensemble. We combine forecasts of the ten best models by equally weighting their outcomes.

3.3.4 Hyperparameters

In the previous subsections, we explain the setup of the FFNN algorithm and also note some of the hyperparameter settings. To gain more insight in the hyperparameter, we provide an overview in Appendix A.3. The FFNN algorithm is written in Python. Additional functions and used packages are also included in Appendix A.3.

3.4 Model combinations

In this section, we discuss the collection of model combinations that will be used in this research. Timmermann (2006) provides a comprehensive outlook on forecast combinations. He finds that simple forecast combinations often outperform more complicated combinations. Therefore, we consider the simple average model combination (SCOMB):

$$\hat{f}_{i,t} = \frac{1}{M} \sum_{m=1}^{M} \hat{f}_{i,m,t}, \tag{11}$$

where $\hat{f_{i,m,t}}$ indicates the RV forecast for stock i of model m at time t and $M = 7$ indicating the total amount of forecast models.

3.4.1 peLASSO forecast combination

Diebold and Shin (2019) propose a forecast combination using the abilities of both shrinkage models and simple-average combinations. Their method relates to the idea of eliminating 'bad' models and averaging over the remaining sufficient models. This is called the partially egalitarian LASSO forecast combination (peLCOMB):

In this research, we use a two-step approach to the peLASSO forecast combination as described by Diebold and Shin (2019). First, we fit a LASSO model on the forecasts. Second, we find which LASSO parameters are nonzero and set all these parameters such that they have equal weights.

First, we forecast using all the individual models on the combination set. Thus, we obtain seven different series containing 847 RV forecasts. Second, we fit a LASSO regression on the model forecasts and the actual RV values in the combination set. A LASSO regression is a penalized linear regression where we minimize the squared estimation error plus a penalty term (Tibshirani, 1996). We estimate the LASSO regression by minimizing the following function (Friedman et al., 2010):

$$\min \left[\frac{1}{2S} \sum_{i=1}^{S} \left(\left(RV_{i,t} - \beta_{i,0} - \sum_{m=1}^{M} \beta_{i,m} \hat{f}_{i,m,t} \right)^2 + \lambda \sum_{j=0}^{M} |\beta_{i,j}| \right] \right], \tag{12}$$

where $\hat{f}_{i,m,t}$ is the RV forecast of stock i by model m at time t , $M = 7$ is the total amount of models, $S = 25$ is the total amount of stocks, and $\lambda = 0.1$ is the shrinkage parameter.

We use the `glmnet` package in R to estimate the LASSO parameters. The `glmnet` algorithm computes the parameters by using a coordinate descent algorithm (Friedman et al., 2010). By fitting the LASSO regression, we now have some parameters that are set to zero and some that are nonzero. In the third step, we set all the nonzero LASSO parameters - combination weights

- to equal weight. Resulting, we eliminate some models in our model combination by setting their weight to zero but maintaining equal weight over the remaining models. We obtain the following for the partially egalitarian LASSO model combination (peLCOMB):

$$\hat{f}_{i,t} = \sum_{m=1}^{M} w_{i,m} \hat{f}_{i,m,t}, \tag{13}$$

where $w_{i,m}$ is the weight from the two-step peLASSO for stock i for model m.

3.5 Volatility quantile-based forecast combination

The volatility quantile-based forecast combination (VQCOMB) aims to segment RV into q quantiles based on their volatility. Next, a peLASSO forecast combination is made for each of these quantiles. Consequently, we obtain q weight vectors. The VQCOMB is a method that aims to combine the forecast accuracy of multiple volatility forecasting models and cherry-pick models on each volatility quantile. Christensen et al. (2022) shows that a neural network is accurate in mid to high volatility and the HARX model performs well in low volatility. This presents the possibility of combining models and weighing them differently according to the expected volatility quantile.

To construct the volatility quantiles, we use IV. Since this variable captures the expected volatility, we use it to forecast the volatility quantile. The quantiles are constructed using data from the training set. We order the IV and construct $q = 10$ volatility quantiles based on their order such that the first $N_{train}/10$ IVs correspond to quantile $q = 1$.

The HAR models and FFNN are fitted on the training set and forecast one day-ahead RV on the combination set. A forecast $R\hat{V}_t 1$ is labeled according to the volatility quantile q it got assigned by IV_{t-1}. We filter the RV forecasts into 10 quantiles for each stock using the labels. For each collection of forecasts, we fit a peLASSO forecast combination as described in Section 3.4.1. If a quantile contains only one observation or all peLASSO parameters are zero, the VQCOMB weight for that quantile is set to equal weights (SCOMB). Thus, for each stock, we obtain $w_{i,q}$ where $q = 1, ..., 10$. We construct the model forecast of VQCOMB as follows:

$$\hat{f}_{i,t} = \sum_{m=1}^{M} w_{i,q,m} \hat{f}_{i,m,t} [min_q < IV_{t-1} \leq max_q], \tag{14}$$

where min_q and max_q are the minimum and maximum IV values within the volatility quantile q.

3.6 Model evaluation

To evaluate forecast results by the forecast models and model combinations, we use the relative mean squared error (relMSE) and QLIKE function. The MSE is a widely used model forecast evaluation measure. However, we use the relMSE, because it allows us to look at how models perform relative to each other. The QLIKE function is an evaluation measure specific to volatility forecasting (Bollerslev et al., 1994).

The relMSE is calculated as follows:

$$\text{relMSE}_{b,m} = \frac{1}{S}\sum_{i=1}^{S}\frac{\text{MSE}_{i,m}}{\text{MSE}_{i,b}}, \tag{15}$$

where $S = 25$ are the total amount of stocks, $\text{MSE}_{i,m}$ is the MSE for model m of stock i, the $\text{MSE}_{i,b}$ is the MSE for the benchmark model b of stock i. We compute the relMSE for both the combination and test set.

We formulate the QLIKE of a model m for a stock i as follows:

$$\text{QLIKE}_{m,i} = \frac{1}{n}\sum_{t=1}^{\text{n}}\left(\log\left(\hat{f}_{m,i,t}^{2}\right) + RV_{i,t}^{2}\hat{f}_{m,i,t}^{-2}\right), \tag{16}$$

where n is the total number of evaluated observations. To get the actual Q-function for model m, we average over all the stocks $\text{QLIKE}_m = \frac{1}{S}\sum_{i=1}^{S}\text{QLIKE}_{m,i}$. Patton (2011) finds that only the MSE and QLIKE are robust to noise in assessing volatility forecasts. The MSE is sensitive to outliers of forecast errors in the right tail. The QLIKE function is robust to outliers in the right tail, but is extremely sensitive to forecast errors in the left tail - so for negative forecast errors.

To further evaluate the forecast performance, we also use the one-sided Diebold-Mariano (DM) test to verify whether a model significantly outperforms another model. We use `dm.test` from the `forecast` package in R to compute the p-values for the DM tests. The `dm.test` function is based on the theoretical framework proposed by Harvey et al. (1997).

4 Results

In this section, we present the results of our empirical analysis of the forecast performance of various models for predicting volatility. We compare the performance of the HAR models, FFNN, and combination models using metrics such as relative MSE, DM tests, and the QLIKE measure. Additionally, we analyze the performance of the models across different volatility regimes and assess the difference in forecasting performance between simple and complex combination models.

Table 1: Relative MSE of RV forecasts in test set

	HAR	logHAR	HARQ	HARQF	SHAR	HARX	FFNN	SCOMB	peLCOMB	VQCOMB
HAR		1.327	0.950	0.816	0.941	0.848	1.238	***0.867***	0.897	**0.896**
logHAR	**0.754**		**0.716**	**0.615**	0.709	**0.639**	0.933	***0.654***	**0.676**	***0.675***
HARQ	1.052	1.396		0.858	0.990	0.892	1.303	0.912	0.944	0.942
HARQF	1.226	1.627	1.165		1.153	1.039	1.518	1.063	1.100	1.098
SHAR	1.063	1.410	1.010	0.867		**0.901**	1.316	**0.922**	0.954	0.952
HARX	1.180	1.565	1.121	0.962	1.110		1.461	1.023	1.058	1.057
FFNN	0.808	1.071	0.768	0.659	0.760	0.685		0.700	0.724	0.723
SCOMB	1.153	1.530	1.096	0.941	1.085	0.978	1.428		1.034	1.033
peLCOMB	1.115	1.479	1.060	0.909	1.049	0.945	1.380	0.967		0.999
VQCOMB	1.116	1.481	1.061	0.911	1.050	0.946	1.382	0.968	1.001	

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

First, we examine the performance of the SCOMB compared to the HAR model. Table 1 reveals that the SCOMB reduces the MSE by 13.3% relative to the HAR model. The DM tests indicate a 5% significance level for more than 50% of the stocks, specifically 17 out of 25. This means that the null hypothesis that the models have equally predictive power is rejected for more than half of the stocks. Furthermore, in Appendix A.4.1 Table 8, we observe that the MSE of the SCOMB is consistently smaller than the MSE of the HAR model for each individual stock. However, in Table 9, we find that the average QLIKE is only 2.7% smaller for the SCOMB compared to the HAR. Still, we can conclude that the SCOMB significantly outperforms the HAR model.

We also compare the SCOMB with other models. The SCOMB outperforms the logHAR and SHAR models, as shown in Table 1 by the lower MSE values (35.6% and 7.8% lower, respectively). Moreover, the DM tests indicate a 5% (logHAR) and 10% (SHAR) significance level for more than 50% of the stocks. Similarly, in Appendix A.4.1 Table 9, the SCOMB has a lower average QLIKE compared to these models. However, for the HARQ(F), HARX, and FFNN, the results are inconclusive. Although the MSE of the SCOMB is lower than that of HARQF and HARX, the DM tests do not show significance. In addition, for the FFNN, the average QLIKE shows a decrease of 0.8% indicating equally predictive power. Thus, the SCOMB improves on the HAR, logHAR, and SHAR, but is inconclusive for the HARQ(F), HARX, and FFNN.

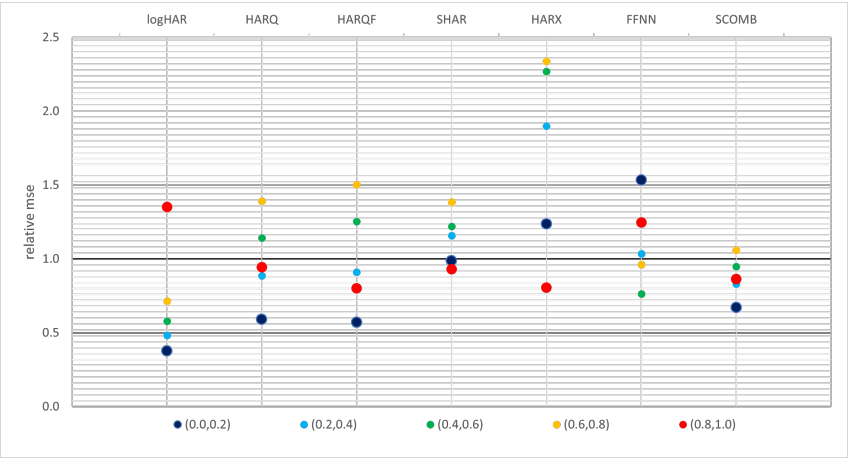
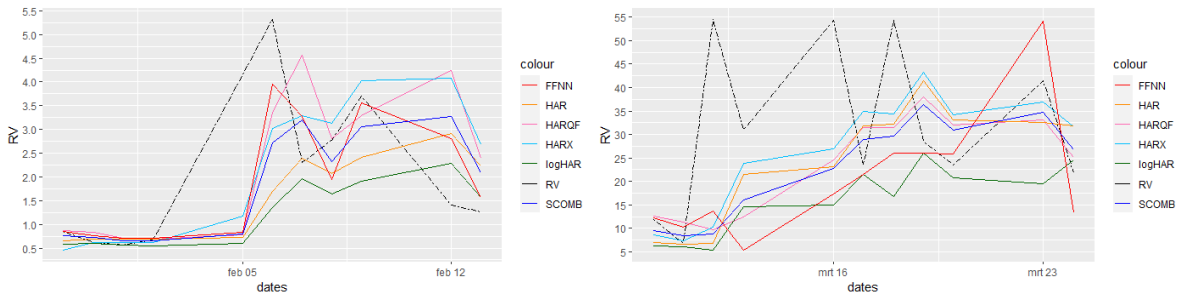


Figure 2: The figure presents the relative Mean Squared Error (MSE) of RV forecasts for each model, excluding peLCOMB and VQCOMB, in the test set across different volatility regimes. The volatility regimes are defined based on observed RV values and are organized into five distinct categories. Each regime represents a quintile of RV forecasts, with the first regime containing the lowest RV values and the last regime containing the highest RV values. The ordering of the quintiles follows the actual observed RV values. Hence, (0.0,0.2) contains forecasts for RVs that correspond to the 20% lowest RVs in the test set.

In evaluating the forecasting performance of various models under different volatility regimes, we first focus on low volatility. Figure 2 shows the relative MSE of the models to HAR in the test set. Actual values are provided in Appendix A.4.1 tables 7 and 6. We find that the logHAR stands out with an MSE reduction of 65% compared to the HAR. This is most likely due to the hesitant behavior of the logHAR to respond to changes in the RV. Figure 3a illustrates this by showing that the logHAR is the most conservative in responding to an increase in RV. Moreover, the HARQ and HARQ reduce MSE by 51% and 55% respectively. As figure 3a shows that the HARQF responds quickly to the increase in RV, we would expect that the model would underperform in a low volatility regime. However, the HARQ and HARQF are set up such that they allow for greater persistence when measurement errors are small and smaller persistence when measurement errors are large. In a low volatility regime, measurement errors are small, so the HARQ(F) is not responsive to changes in RV. FFNN has the highest relative MSE with an increase of 73%. The FFNN is likely too sensitive to changes in RV such that it overestimates RV in low volatility. The responsiveness of FFNN is further supported in figure 3a. Thus, in a low volatility regime, the most conservative model proves to be the most accurate.



(a) January 30, 2018 - February 13, 2018 (11 observations) (b) March 10, 2020 - March 24, 2020 (11 observations)

Figure 3: These figures display line graphs of the RV and the RV forecasts of six models for Coca-Cola Co.(KO) on two specific subsamples in the test set. KO is selected as example, because the MSE and QLIKE of the six selected models correspond with the average MSE and QLIKE over all the stocks. The forecast models include: HAR, logHAR, HARQF, HARX, FFNN, and SCOMB. For robustness, we provide line graphs for two different stocks for the same time periods in Appendix A.5.1.

Next, we compare models in high volatility. The HARQF and HARX show the largest reduction in MSE (14% and 21% respectively). As previously discussed, the HARQF is able to switch persistence of historical volatility when measurement errors vary. This makes it an agile model to respond to increases in volatility. Comparing this to the MSE reduction of merely 7% for the HARQ. The weekly and monthly time-varying parameter of the HARQF give more accurate results in high volatility as opposed to only the daily time-varying parameters for the HARQ. HARX ensures the greatest MSE reduction. This is most likely due to the IV variable

which has an forward-looking ability. This forward-looking ability is illustrated in figure 3a as the HARX increases in RV before any other model does. Underperforming models in the high volatility regime include the logHAR and FFNN with an MSE increase of 22% and 12%. Though FFNN shows adequate responsiveness in figure 3a, MSE indicates inaccuracy. This is congruent with figure 3b, where FFNN portrays ineffect forecasting. Overall, figure 3b illustrates the inability of models to forecast RV in extreme volatility. This is universal, even for the HARX.

In the mid volatility regimes, logHAR and FFNN are most accurate. Here - on average - logHAR reduces MSE with 40% whereas FFNN reduces MSE with 20%. The conservative nature of the logHAR makes it also an adequate model in mid volatility regimes. Findings indicating accuracy in mid volatility regimes are in accordance with previous literature (Christensen et al., 2022). Notably, is the extreme inflation of MSE from the HARX in mid volatility regimes. This varies from an increase in MSE of approximately 80-130%. The HARX has been shown to be inaccurate in mid volatility regimes with inflation of MSE around 10-60% (Christensen et al., 2022). Hence, our results deviate from previous findings. The large MSE values could be explained by the sensitivity of HARX to IV. Increased values of IV could trigger an supposedly unnecessary reaction to a high volatility expectation. This in term creates inaccurate forecasts. SCOMB performs steadily in mid volatility with relative MSE values not deviation much, indicating robustness.

Table 1 shows that SCOMB has the lowest MSE compared to peLCOMB and VQCOMB. Also in Appendix A.4.1 Table 6, we find that the SCOMB performs better on all volatility regimes. This is due to the weighting of peLCOMB and VQCOMB. The weight of the model combinations is estimated on the combination set. But since the forecast models have different results in the combination set, the weights are not fitted right for the test set. Appendix A.4.1 Table 5 shows contradictory results for the logHAR and FFNN compared to the test set. The peLCOMB mostly selects the HARX and FFNN. Since these models are inaccurate in the test set, this explains the higher MSE for the peLCOMB. As for the VQCOMB, its weights are more evenly distributed. But again, the heavy weighting of the HARX and FFNN makes VQCOMB noncompetitive. Moreover, there is a major critique of whether the forward-looking IV can segment the following RV value in the right volatility quantile. Concluding, a simple model combination is desirable to a more complex model combination since the performance of volatility models can vary significantly between sample sets.

Table 1 reveals that the SCOMB model outperforms both peLCOMB and VQCOMB in terms of mean squared error (MSE). Additionally, in Appendix A.4.1, Table 6 demonstrates that SCOMB consistently performs better across all volatility regimes. These findings can be

explained by the weighting methodology employed in the model combinations. It is worth noting that the weights of the model combinations are estimated on the combination set. Subsequently, since the forecast models yield different results within the combination set, the weights may not be appropriately fitted for the test set. As shown in Appendix A.4.1, Table 5, the results for logHAR and FFNN contradict those obtained from the test set. Notably, peLCOMB predominantly selects HARX and FFNN, which are inaccurate in the test set, resulting in a higher MSE for the peLCOMB. Similarly, VQCOMB distributes its weights more evenly, but the substantial weighting of HARX and FFNN lowers the competitiveness of VQCOMB. For the VQCOMB the critical concern arises that the IV cannot accurately label volatility regimes resulting in an ineffective weighting scheme. Consequently, we find that a simpler model combination approach is preferable over a more complex one, as the performance of volatility models can exhibit significant variation across different sample sets.

5 Conclusion

This research aims to answer whether a combination of HAR models and an 'off-the-shelf' neural network can improve one day-ahead RV forecasts. Furthermore, we look at the model forecasting performance in various volatility regimes. And lastly, we research whether a simple or complex model combination is desirable when forecasting RV.

The SCOMB model outperforms the HAR model with a 13.3% reduction in MSE and significant ($p < 0.05$) DM test results for more than half of the stocks. It also improves on the logHAR and SHAR models, showing lower MSE values (35.6% and 7.8% lower respectively) and significant (5% and 10%) DM test results. However, its performance is inconclusive compared to the HARQ(F), HARX, and FFNN models. In low volatility regimes, the logHAR shows a significant 65% MSE reduction, while the FFNN overestimates volatility with a 73% relative MSE increase. In high volatility regimes, the HARQF and HARX models have substantial MSE reductions (14% and 21% respectively), while the logHAR and FFNN models underperform. In mid volatility regimes, the logHAR and FFNN models show higher accuracy, reducing MSE by 40% and 20% respectively, whereas the HARX model has an unexpectedly large inflation of MSE. The SCOMB model outperforms peLCOMB and VQCOMB in terms of MSE and across all volatility regimes, which shows the importance of a simpler model combination approach. A simple model combination is preferable since the model forecast performance can change quickly between sample sets. Overall, SCOMB proves to be a robust model to forecast RV, since model performance varies across volatility regimes.

The research has several limitations that need to be considered. First and foremost, the RV

forecasts of the FFNN are less accurate compared to the similar NN^3_{10} by Christensen et al. (2022). The relative MSE on the same sample set is 0.947 for the FFNN and 0.898 for the NN^3_{10}. This indicates differences in the models, probably due to hyperparameter settings. Consequently, the relevance of the model combinations presented in this research is mostly due to the robustness they provide. Whereas the aim of the research was to investigate whether a combination of HAR and FFNN could complement each other. Since the FFNN in the test were inaccurate, the model combinations mostly depended on the HAR models. Second, volatility quantile forecasting using IV is not efficient in forecasting the following volatility regimes. This makes the VQCOMB a relatively irrelevant model combination. However, the underlying principle is interesting and could still be an area of future research. Third, due to time constraints, the RV forecast of the logHAR was not transformed appropriately. According to Equation 5, an inflation factor due to the transformation of the error term in Equation 4 is needed. This lacks in our transformation of the logHAR. Fourth, the QLIKE and MSE are limited metrics in assessing volatility forecasts. The MSE is highly influenced by outliers that result in high positive forecast errors. On the other hand, the QLIKE function is robust to outliers for high positive forecast errors but is susceptible to outliers with high negative forecast errors - overestimation of RV. It is therefore important to regard both metrics.

For future research, we recommend research outlining the practical implementation of implied volatility can be useful. Since implied volatility data is hard to come by and calculations are complex and computationally demanding, an easy-to-use implementation can be convenient for researchers who are not interested to spend a lot of time on these issues. Moreover, a practical outline that implied volatility data can be discarded and what data contains important information is also interesting for further research. Furthermore, we suggest further research in studying extremely high volatility cases and how to forecast this. Current models are ineffective in handling these cases. Lastly, further research into neural networks and the promising LSTM RNN are the future of volatility forecasting.

In conclusion, a combination of HAR models and a neural network is able to improve the benchmark HAR, SHAR, and logHAR models, but fails to significantly outperform the other models. That said, the simple SCOMB combination model is able to provide robust forecasts in various volatility regimes and beats more complex combination models while maintaining computational simplicity.

References

Aït-Sahalia, Y., Fan, J., & Li, Y. (2013). The leverage effect puzzle: Disentangling sources of bias at high frequency. *Journal of Financial Economics*, *109*(1), 224–249. https://doi.org/10.1016/j.jfineco.2013.02.018

Barndorff-Nielsen, O. E., & Shephard, N. (2002). Estimating quadratic variation using realized variance. *Journal of Applied Econometrics*, *17*(5), 457–477. https://doi.org/10.1002/jae.691

Bollerslev, T., Engle, R. F., & Nelson, D. B. (1994). Chapter 49 arch models. In *Handbook of econometrics* (pp. 2959–3038). Elsevier. https://doi.org/10.1016/s1573-4412(05)80018-2

Bollerslev, T., Patton, A. J., & Quaedvlieg, R. (2016). Exploiting the errors: A simple approach for improved volatility forecasting. *Journal of Econometrics*, *192*(1), 1–18. https://doi.org/10.1016/j.jeconom.2015.10.007

Bucci, A. (2020). Realized volatility forecasting with neural networks. *Journal of Financial Econometrics*, *18*(3), 502–531. https://doi.org/10.1093/jjfinec/nbaa008

CBOE. (2022). Cboe volatility index® mathematics methodology.

Christensen, K., Siggaard, M., & Veliyev, B. (2022). A machine learning approach to volatility forecasting. *Journal of Financial Econometrics*. https://doi.org/10.1093/jjfinec/nbac020

Corsi, F. (2008). A simple approximate long-memory model of realized volatility. *Journal of Financial Econometrics*, *7*(2), 174–196. https://doi.org/10.1093/jjfinec/nbp001

Demeterfi, K., Derman, E., Kamal, M., & Zou, J. (1999). More than you ever wanted to know about volatility swaps.

Diebold, F., & Shin, M. (2019). Machine learning for regularized survey forecast combination: Partially-egalitarian LASSO and its derivatives. *International Journal of Forecasting*, *35*(4), 1679–1691. https://doi.org/10.1016/j.ijforecast.2018.09.006

Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, *33*(1). https://doi.org/10.18637/jss.v033.i01

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning* [http://www.deeplearningbook.org]. MIT Press.

Harvey, D., Leybourne, S., & Newbold, P. (1997). Testing the equality of prediction mean squared errors. *International Journal of Forecasting*, *13*(2), 281–291. https://doi.org/10.1016/s0169-2070(96)00719-4

Jiang, G. J., & Tian, Y. S. (2005). The model-free implied volatility and its information content. *Review of Financial Studies*, *18*(4), 1305–1342. https://doi.org/10.1093/rfs/hhi027

Kambouroudis, D. S., McMillan, D. G., & Tsakou, K. (2021). Forecasting realized volatility: The role of implied volatility, leverage effect, overnight returns, and volatility of realized volatility. *Journal of Futures Markets*, *41*(10), 1618–1639. https://doi.org/10.1002/fut.22241

Kleen, O., & Tetereva, A. (2022). A forest full of risk forecasts for managing volatility. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.4161957

Patton, A. J. (2011). Volatility forecast comparison using imperfect volatility proxies. *Journal of Econometrics*, *160*(1), 246–256. https://doi.org/10.1016/j.jeconom.2010.03.034

Patton, A. J., & Sheppard, K. (2015). Good volatility, bad volatility: Signed jumps and the persistence of volatility. *Review of Economics and Statistics*, *97*(3), 683–697. https://doi.org/10.1162/rest_a_00503

Rahimikia, E., & Poon, S.-H. (2020). Machine learning for realised volatility forecasting. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3707796

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, *58*(1), 267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

Timmermann, A. (2006). Chapter 4 forecast combinations. In *Handbook of economic forecasting* (pp. 135–196). Elsevier. https://doi.org/10.1016/s1574-0706(05)01004-9

Wang, B., Zhang, Y., Zhang, H., Meng, Q., Ma, Z.-M., Liu, T.-Y., & Chen, W. (2022). Provable adaptivity in adam. https://doi.org/10.48550/ARXIV.2208.09900

Zhang, C., Zhang, Y., Cucuringu, M., & Qian, Z. (2023). Volatility forecasting with machine learning and intraday commonality. *Journal of Financial Econometrics*. https://doi.org/10.1093/jjfinec/nbad005

A Appendix

A.1 Variables

Table 2: Variable description

Name	Description	Formula
RV	Realized variance	$\text{RV}_{i,t} = \sum_{j=1}^{n}(r_{i,j})^2$
RQ	Realized quarticity	$\text{RQ}_{i,t} = \frac{n}{3}\sum_{j=1}^{n}(r_{i,})^4$
RV+	Realized positive semivariance	$\text{RV}_{i,t}^{+} = \sum_{j:r_{i,j}>0}(r_{i,j})^2$
RV-	Realized negative semivariance	$\text{RV}_{i,t}^{-} = \sum_{j:r_{i,j}<0}(r_{i,j})^2$
VIX	CBOE volatility index*	$\sigma^2 = \frac{2}{T}\sum_i \frac{\Delta K_i}{K_i^2}e^{RT}Q(K_i) - \frac{1}{T}[\frac{F}{K_0} - 1]^2$
IV	Implied volatility**	-
MOM	1-week momentum	$\text{MOM}_{i,t} = \frac{(X_{i,t} - X_{i,t-5})}{X_{i,t-5}}$
TV	dollar trading volume***	-

Notes: This table shows the variable name and description with the corresponding formula. *Denotes the CBOE volatility index using the calculation provided by Demeterfi et al. (1999). Regard the document provided by CBOE (2022) for more information on the calculation procedure. ** The implied volatility is obtained by taking the median of the implied volatilities of a stock's option pricing through OptionMetrics. Regard the data section for more information. ***Dollar trading volume denotes the cumulative sum of dollars spent on trades on an asset during one trading day.

Table 3: Descriptive statistics of the variables

Sample set	Volatility		RVD	RVW	RVM	RQ	RV+	RV-	VIX	MOM	TV	IV
Train	26.883	Mean	2.93	2.35	2.28	1.05E+02	1.48	1.46	21.90	1570.00	4.11E+07	11.00
		Min.	0.15	0.20	0.33	3.41E-02	0.05	0.05	9.89	-2.39	3.72E+06	0.04
		Max.	149.00	52.90	28.50	8.61E+04	69.30	62.70	80.90	0.25	2.74E+08	0.62
		Std. deviation	5.51	3.67	3.07	1.85E+03	2.85	2.67	9.57	0.04	2.51E+07	0.07
Validation	14.201	Mean	0.82	0.66	0.64	3.66E+00	0.43	0.42	14.10	0.00	2.63E+07	0.06
		Min.	0.16	0.21	0.30	3.40E-02	0.06	0.05	10.30	-7.45	7.41E+06	0.04
		Max.	6.07	2.27	1.28	3.60E+02	3.68	3.77	22.70	0.08	1.38E+08	0.10
		Std. deviation	0.58	0.31	0.21	2.01E+01	0.34	0.36	1.97	0.02	1.39E+01	0.01
Combination	15.743	Mean	0.10	0.80	0.77	2.91E+01	0.52	0.51	14.60	0.00	1.66E+07	0.06
		Min.	0.13	0.17	0.24	1.98E-02	0.05	0.05	9.14	-0.12	4.36E+06	0.04
		Max.	43.40	11.40	3.86	1.97E+04	23.00	17.00	40.70	0.12	9.12E+07	0.16
		Std. deviation	1.84	0.94	0.58	6.82E+01	0.98	0.83	4.17	0.03	8.35E+06	0.02
Test	23.917	Mean	2.35	1.88	1.81	9.49E+01	1.25	1.24	20.20	0.00	1.59E+07	0.11
		Min.	0.19	0.21	0.33	4.93E-02	0.07	0.07	9.15	-0.22	4.21E+06	0.04
		Max.	79.60	45.80	25.60	2.26E+04	46.40	42.10	82.70	0.24	8.03E+07	0.75
		Std. deviation	5.29	3.85	3.10	9.41E+02	2.98	2.78	8.90	0.04	8.27E+06	0.07

*Notes:*This table contains the descriptive statistics of the variables contained within the dataset. Variables descriptions can be seen in Table 2. The volatility denotes the annualized realized volatility. All metrics are calculated over all stocks and then averaged.

A.3 Hyperparameters tuning on feed-forward neural network

Table 4: Hyperparameter settings for the FFNN

Description	Python function	Hyperparameters
Standardization of input variables	sklearn.preprocessing.StandardScaler	
Feed-forward neural network	keras.layer.Sequential	
Hidden layers	keras.layers.Dense	x = {64, 32, 16, 1}, activation = 'linear'
Activation function	keras.layers.LeakyReLu	alpha = 0.1
Dropout function	keras.layers.Dropout	0.2
Parameter optimization	keras.optimizers.Adam	learning_rate = 0.001 * np.random.uniform(0.5, 1.5)
Early stopping of epochs	keras.callbacks.EarlyStopping	patience = 100
Maximum number of epochs		epochs = 500
Batch size for model fitting		batch_size = 32
Number of models in ensemble		n_ensemble = 10
Total number of models		n_models = 100
Kernel initializer	keras.initializers.GlorotNormal	

Notes: This table contains the hyperparameter settings and configurations of the network architecture. As ensemble method we employ an equal weighting scheme for the ten best models, ranked by their MSE.

A.4 Results

A.4.1 Combination set (replication of Christensen et al. (2022))

In table 5 we provide the relative MSE of all models in the combination set. These MSEs should correspond to values for the test set in the study by Christensen et al. (2022).

Table 5: Relative MSE in combination set

	HAR	logHAR	HARQ	HARQF	SHAR	HARX	FFNN
HAR		0.976	1.004	1.009	0.984	0.982	***0.947***
logHAR	1.025		1.029	1.034	1.009	1.007	0.971
HARQ	0.996	0.972		1.005	0.980	0.978	***0.943***
HARQF	0.991	0.967	0.995		0.975	0.974	***0.939***
SHAR	1.016	0.991	1.020	1.025		0.998	***0.962***
HARX	1.018	0.993	1.022	1.027	1.002		**0.964**
FFNN	1.056	1.030	1.060	1.066	1.039	1.037	

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

Figure 4 shows the relative MSE performance of all models in the combination set for five volatility regimes. This figure should resemble the figure on relative MSE of models in various volatility regimes in Christensen et al. (2022). We note that Christensen et al. (2022) select different quintiles for their volatility regimes.

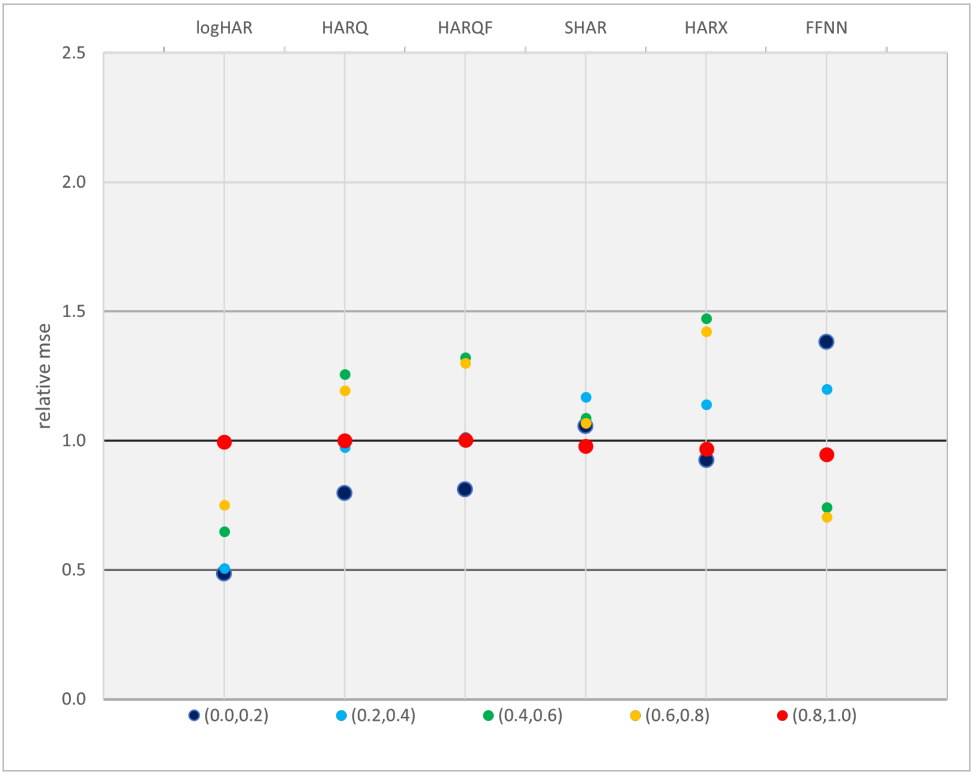


Figure 4: The figure presents the relative Mean Squared Error (MSE) of RV forecasts for each model in the combination set across different volatility regimes. The volatility regimes are defined based on observed RV values and are organized into five distinct categories. Each regime represents a quintile of RV forecasts, with the first regime containing the lowest RV values and the last regime containing the highest RV values. The ordering of the quintiles follows the actual observed RV values. Hence, (0.0,0.2) contains forecasts for RVs that correspond to the 20% lowest RVs in the combination set.

A.4.2 Additional results from the test set

Table 6 identifies the forecasting performance of forecasting models on the test set for a given volatility regime relative to the HAR. This ensures easy comparison between models.

Table 6: Relative MSE for 10 volatility regimes in the test set

	rv	logHAR	HARQ	HARQF	SHAR	HARX	FFNN	SCOMB	peLCOMB	VQCOMB
1	0.35	0.35	0.49	0.45	0.97	1.04	1.73	0.63	0.75	0.89
2	0.53	0.40	0.67	0.67	1.00	1.39	1.38	0.70	0.84	0.94
3	0.68	0.46	0.81	0.85	1.11	1.79	1.11	0.80	0.96	1.14
4	0.83	0.50	0.95	0.97	1.20	2.00	0.97	0.85	0.99	1.11
5	1.03	0.55	1.05	1.15	1.28	2.16	0.79	0.92	1.07	1.18
6	1.27	0.60	1.22	1.34	1.17	2.36	0.74	0.97	1.08	1.19
7	1.60	0.69	1.34	1.44	1.28	2.53	0.82	1.01	1.18	1.17
8	2.12	0.73	1.42	1.54	1.45	2.23	1.04	1.09	1.17	1.14
9	3.13	0.84	1.36	1.46	1.34	1.90	1.45	1.05	1.20	1.03
10	11.87	1.22	0.93	0.86	0.97	0.79	1.12	0.84	0.93	0.87

Notes: This table shows the MSE of a forecast model on the test set relative to the HAR for various volatility regimes. Each regime represents a decile of RV forecasts, with the first regime containing the lowest RV values and the last regime containing the highest RV values. The ordering of the deciles follows the actual observed RV values. Hence, decile 10 contains forecasts for RVs that correspond to the 10% highest RVs in the test set. The RV column indicates the average RV in that volatility regime.

Table 7 identifies the forecasting performance of forecasting models on the test set for a given volatility regime. This table shows the absolute MSE values for each volatility regime. We observe that the MSE in the highest volatility decile is disproportionately high compared to the other deciles.

Table 7: MSE for 10 volatility regimes in the test set

	rv	HAR	logHAR	HARQ	HARQF	SHAR	HARX	FFNN	SCOMB	peLCOMB	VQCOMB
1	0.355	0.187	0.065	0.092	0.084	0.181	0.195	0.322	0.118	0.141	0.167
2	0.530	0.228	0.091	0.153	0.154	0.229	0.318	0.314	0.161	0.191	0.215
3	0.675	0.291	0.135	0.237	0.246	0.323	0.520	0.322	0.234	0.280	0.332
4	0.834	0.320	0.159	0.305	0.310	0.384	0.642	0.311	0.273	0.317	0.357
5	1.026	0.429	0.234	0.451	0.494	0.547	0.925	0.339	0.393	0.460	0.508
6	1.266	0.530	0.320	0.644	0.708	0.622	1.250	0.392	0.515	0.571	0.630
7	1.598	0.639	0.438	0.853	0.923	0.815	1.616	0.527	0.648	0.755	0.748
8	2.118	1.118	0.819	1.594	1.719	1.617	2.492	1.164	1.214	1.305	1.271
9	3.130	2.470	2.081	3.368	3.614	3.302	4.691	3.570	2.598	2.960	2.541
10	11.868	97.149	118.562	90.507	83.749	94.073	77.143	108.804	81.426	90.535	84.151

Notes: This table shows the MSE of a forecast model for various volatility regimes. Each regime represents a decile of RV forecasts, with the first regime containing the lowest RV values and the last regime containing the highest RV values. The ordering of the deciles follows the actual observed RV values. Hence, decile 10 contains forecasts for RVs that correspond to the 10% highest RVs in the test set. The RV column indicates the average RV in that volatility regime.

Table 8 presents the MSE for each model for a particular stock in the test set. The column on the left shows the stock's ticker.

Table 8: MSE per stock in the test set

	HAR	logHAR	HARQ	HARQF	SHAR	HARX	FFNN	SCOMB	peLCOMB	VQCOMB
AAPL	8.34	8.59	8.07	8.36	8.12	7.96	7.95	7.90	8.47	8.02
AXP	12.27	13.46	10.53	10.62	12.72	9.55	9.14	9.92	12.29	9.92
BA	86.73	143.16	83.21	49.30	70.29	72.37	138.67	74.65	69.81	74.56
CAT	7.92	9.54	7.57	7.57	8.42	7.20	6.56	7.01	8.17	7.05
CSCO	7.11	8.40	7.07	6.78	7.06	6.17	5.40	6.19	6.45	6.28
CVX	11.98	14.34	10.74	10.53	12.17	9.62	14.61	9.84	10.09	9.37
DIS	8.24	9.95	7.21	7.10	7.94	6.83	9.58	6.93	7.83	8.91
GE	23.40	25.84	23.48	23.75	23.79	23.87	22.85	21.48	22.13	21.47
GS	8.40	8.99	7.33	8.04	9.53	8.03	6.34	5.88	6.34	6.09
HD	11.40	13.10	10.90	10.61	8.84	6.49	9.91	8.56	8.23	8.61
IBM	4.16	5.31	3.71	3.69	4.49	3.50	3.92	3.76	3.88	3.80
INTC	10.75	14.14	10.86	10.71	10.14	9.76	10.66	10.20	11.41	10.06
JNJ	5.27	6.48	5.25	5.12	5.46	4.99	11.91	4.96	5.34	5.65
KO	5.40	6.94	5.14	5.01	5.16	4.71	5.40	4.92	4.91	4.95
MCD	8.50	11.40	6.60	5.86	7.64	6.00	10.09	6.51	6.65	6.80
MMM	5.48	5.13	3.96	4.00	4.65	4.30	3.98	3.91	4.69	3.77
MRK	6.24	6.73	5.85	5.69	6.06	5.73	7.40	5.40	6.08	5.40
MSFT	6.72	8.00	7.03	6.79	6.50	6.27	7.93	6.56	6.50	6.53
NKE	7.91	10.62	7.66	7.12	8.26	6.13	10.39	7.21	9.26	7.31
PFE	6.31	6.91	6.32	6.23	6.26	5.64	5.48	5.84	5.87	6.00
RTX	15.90	22.74	16.94	13.15	13.89	12.04	22.59	13.60	12.75	13.74
TRV	6.29	7.39	5.61	5.93	6.50	5.40	7.65	5.50	6.51	5.57
VZ	4.35	4.94	4.08	3.93	4.50	3.89	3.91	3.97	4.27	4.03
WMT	4.57	5.21	4.17	4.09	4.38	4.32	6.13	4.34	4.64	4.38
XOM	8.61	10.45	8.51	8.39	12.17	6.99	13.43	8.38	9.60	13.50
Average	11.69	15.51	11.11	9.53	11.00	9.91	14.48	10.14	10.49	10.47

Notes: This table shows the MSE for each model on each stock. All the stocks' tickers are included in the most left column. The bottom row is the average MSE over all stocks. The MSE is calculated over the test set.

Table 9 displays the Qlike for each model for a particular stock in the test set. The column on the left shows the stock's ticker. We note that some Qlike values are extremely high compared to other values. This is due to the sensitivity of Qlike to forecast errors in the left-side - overestimation of RV by forecasting models.

Table 9: QLIKE in the test set

	HAR	logHAR	HARQ	HARQF	SHAR	HARX	FFNN	SCOMB	peLCOMB	VQCOMB
AAPL	2.65	2.90	2.61	2.60	2.62	39938.57	2.62	2.59	2349.21	2.72
AXP	2.93	3.54	3.34	3.93	2.90	4888.84	2.70	2.95	1038.63	2.80
BA	4.04	4.46	4.08	4.55	4.39	22.23	4.30	4.05	4.06	5.76
CAT	3.14	3.34	3.24	3.20	3.14	6.89	3.14	3.14	3.18	6.75
CSCO	2.14	2.37	2.26	2.57	2.18	719.04	2.05	2.11	2.28	2.21
CVX	2.34	2.39	2.47	2.75	9.45	195.56	2.32	2.27	2.37	66.88
DIS	2.64	2.88	2.60	2.84	2.60	6395.32	2.42	2.50	2.63	4525.24
GE	4.47	4.75	4.48	4.55	4.47	4.58	4.50	4.34	4.43	4.36
GS	2.70	2.83	41497.84	499.49	2.70	47466.47	2.73	2.68	2.76	2.73
HD	2.04	2.25	5.84	2.81	1.95	24882.45	1.97	1.94	2.02	2.33
IBM	2.06	2.34	2.40	2.35	2.05	926.57	1.78	2.03	2.10	162.06
INTC	3.41	3.89	3.83	3.88	3.55	3.89	3.23	3.41	3.32	3.44
JNJ	2.48	3.02	2.37	2.27	2.44	13123.50	2.32	2.28	2.50	4708.11
KO	0.99	1.14	1.20	20.82	0.94	809.23	0.87	0.88	0.98	0.91
MCD	1.58	1.89	1.88	885522.40	1.50	764374.60	1.49	1.43	1.85	60485.14
MMM	2.38	2.78	2.35	2.44	6.33	10.60	2.07	2.20	2.66	2.43
MRK	1.99	2.21	1.93	1.94	2.15	3054.87	1.84	1.92	2.01	1.94
MSFT	2.14	2.33	2.20	2.24	2.10	3.20	2.12	2.11	2.16	2.12
NKE	2.19	2.34	2.25	2.33	2.16	3387.56	2.12	2.12	354.49	2324.80
PFE	2.55	2.86	3.25	3.21	2.52	2.64	2.41	2.42	2.49	2.65
RTX	2.56	2.85	3.68	3.21	2.56	586.65	2.47	2.60	2.74	263.03
TRV	1.94	2.16	2.06	42.47	1.99	172198.60	1.96	1.95	41257.22	1.97
VZ	1.24	1.41	1.44	13.56	1.22	1350941.00	1.26	1.19	1.25	1.36
WMT	1.73	2.09	2.09	2.21	1.79	228.26	1.60	1.68	1.64	3.89
XOM	2.21	2.28	2.22	2.36	219.83	630.46	2.29	2.16	2.32	242.31
Average	2.42	2.69	1662.56	35446.28	11.58	97392.06	2.34	2.36	1802.05	2913.12

Notes: This table shows the QLIKE for each model on each stock. All the stocks' tickers are included in the most left column. The bottom row is the average QLIKE over all stocks. The QLIKE is calculated over the test set.

A.5 Robustness checks

A.5.1 Line graphs in the test set

Figure 5 and 6 are line graphs capturing the RV and corresponding RV forecasts for IBM. The figure act as a robustness check for the figures 3a and 3b in the results section. We note in figures 5 and 6 that they also show increased volatility. Since volatility is seen throughout all graphs - also figure 7 and 8 - volatility is market-wide and not firm-specific.

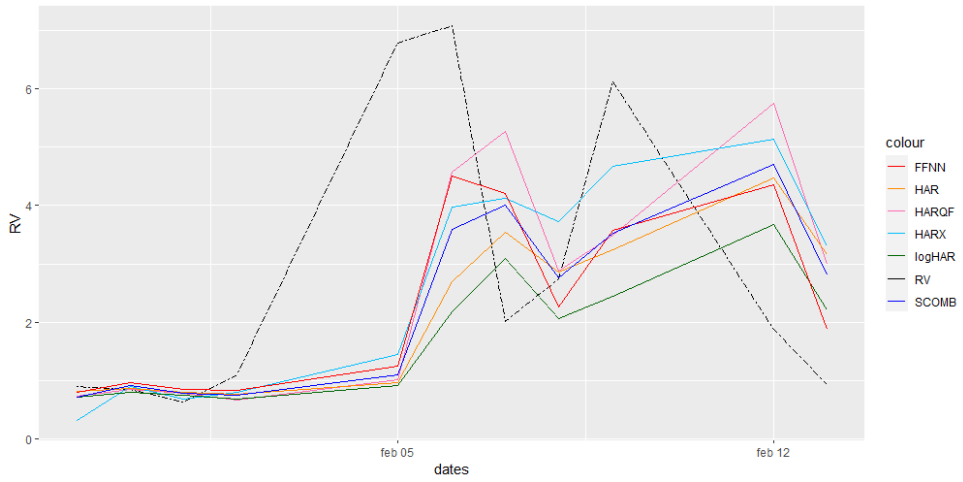


Figure 5: This figure shows the observed RV and RV forecasts of six models for IBM in the test set. The data spans from January 30, 2018 till February 13, 2018 (11 observations).

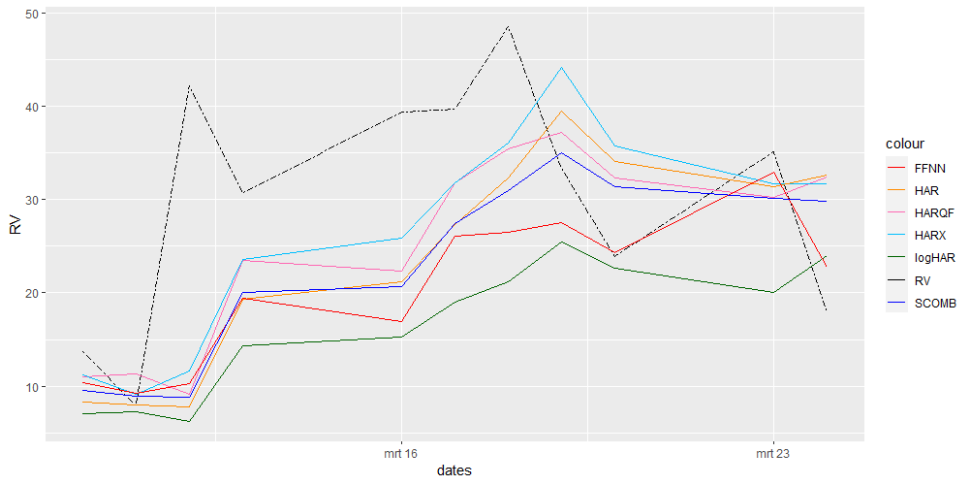


Figure 6: This figure displays the observed RV and RV forecasts of six models for IBM in the test set. The data spans from March 10, 2020 - March 24, 2020 (11 observations).

Figure 5 and 6 show the line graphs for the RV and corresponding RV forecasts for Boeing (BA). Like figures 5 and 6, the figures act as a robustness check for figures 3a and 3b in the results section. We also note here that we observe increased volatility. Especially in figure 8, we observe extremely high volatility. Since Boeing is of course highly correlated to the aviation industry, the turbulent situation around Covid, and the subsequent cancellation of many flights, likely caused the Boeing stock to fluctuate heavily. Moreover, we report that HAR is most responsive in figure 7 to an increase in RV. This is in contrast to previous findings in other figures.

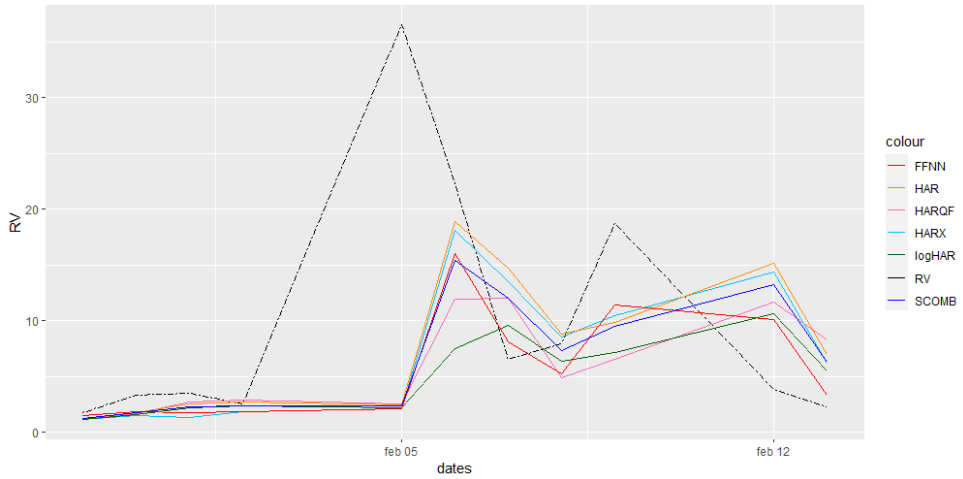


Figure 7: This figure portrays the observed RV and RV forecasts of six models for Boeing (BA) in the test set. The data spans from January 30, 2018, to February 13, 2018 (11 observations).

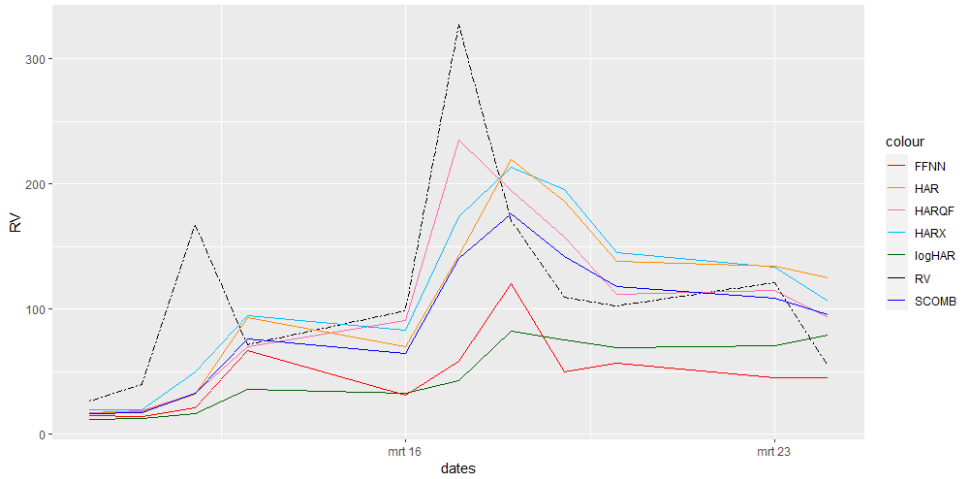


Figure 8: This figure displays the observed RV and RV forecasts of six models for Boeing (BA) in the test set. The data spans from March 10, 2020 - March 24, 2020 (11 observations).

A.6 Additional information on the models

A.6.1 Model combinations

Table 10 exhibits the peLCOMB weights estimated for each stock using the two-stage peLASSO procedure described in subsection 3.4.1. We note the weights that are assigned to the HARX and FFNN are much more predominant than for the other models.

Table 10: Forecast combination weights of peLCOMB

	HAR	logHAR	HARQ	HARQF	SHAR	HARX	FFNN
AAPL							1
AXP							1
BA						0.50	0.50
CAT						0.50	0.50
CSCO						0.50	0.50
CVX						0.50	0.50
DIS						0.50	0.50
GE						0.50	0.50
GS				0.33	0.33		0.33
HD		0.33		0.33		0.33	
IBM						0.50	0.50
INTC						0.50	0.50
JNJ				0.50		0.50	
KO						0.50	0.50
MCD						0.50	0.50
MMM						1	
MRK						0.50	0.50
MSFT						0.50	0.50
NKE							1
PFE						0.50	0.50
RTX						0.50	0.50
TRV							1
VZ		0.50				0.50	
WMT				0.50		0.50	
XOM						0.50	0.50

Notes: This table contains the weights assigned to each model for the peLCOMB by the two-stage peLASSO estimation procedure. The weights are estimated using the forecasts in the combination set.

Table 11 outlines the average VQCOMB weights set for each volatility regime. The VQ-COMB weights are estimated for each stock. Here the weights are averaged over all stocks for each volatility regime.

Table 11: Forecast combination weights per volatility regime of VQCOMB

	HAR	logHAR	HARQ	HARQF	SHAR	HARX	FFNN
1	0.119	0.132	0.219	0.106	0.066	0.199	0.159
2	0.097	0.197	0.117	0.057	0.057	0.337	0.137
3	0.069	0.095	0.055	0.122	0.049	0.349	0.262
4	0.078	0.098	0.091	0.078	0.045	0.298	0.311
5	0.006	0.126	0.106	0.086	0.046	0.326	0.306
6	0.031	0.011	0.045	0.065	0.025	0.505	0.318
7	0.026	0.119	0.026	0.099	0.019	0.252	0.459
8	0.071	0.101	0.021	0.041	0.145	0.335	0.285
9	0.067	0.107	0.073	0.100	0.120	0.333	0.200
10	0.149	0.169	0.109	0.109	0.109	0.149	0.209

Notes: This table contains the average weights assigned to each model for the VQCOMB for each volatility regime. The forecasts in the combination set are labeled using a volatility quantile, indicating to which volatility regime the forecast corresponds. Next, the forecasts are sectioned and peLASSO weights are estimated for each volatility regime. This procedure is repeated for each stock. This table includes the average weights of that procedure for each volatility regime.