Comparison of the predictive performance of Local Linear Forests to the GARCH-MIDAS approach, using financial volatility and macroeconomic variables

Laura van Slooten

480889

Abstract

This paper investigates the forecast performance of GARCH-MIDAS relative to *local linear forests*: a procedure that takes random forests as an adaptive kernel method paired to a local linear regression to capture smooth signals better. In an empirical study with S&P 500 data, Realized volatility is estimated by both models using macroeconomic variables, and the outcomes are compared based on QLIKE losses and mean prediction errors. Furthermore, three additional models are included as competitor models, namely HAR (Heterogeneous autoregression) with and without leverage and Realized GARCH. The findings show that using LLF in combination with all listed macroeconomic variables reduces the prediction losses for 1-day-, 2-weeks-, 1-month-, 2-months-, and 3-months-ahead horizons. Additionally, the SHAP values show the contribution to the prediction for all macroeconomic variables where the VIX, NFCI, and RVol proved to be the most important.

Contents

1 Introduction

This paper focuses on the method described in Friedberg et al., 2020. They introduce local regressions in combination with random forests in order to improve predictive accuracy. This model tackles the main weakness of random forests, now that the prediction function no longer resembles a step function but a smooth curve. The main findings of Friedberg et al. (2020) show that the linear adjustment to the random forest improves the confidence interval coverage as well as the predictive performance. Due to these systematic improvements, a practical application of the local linear forests (LLF) would be to compare the predictive performance on stock market volatility of this method to well-known forecast models.

Volatility modeling using multiple components is widely discussed within financial econometrics (e.g., R. Engle and Lee, 1999). Bollerslev (1986) was the first to introduce the generalized autoregressive conditional heteroskedasticity model (GARCH), and it is beneficial to model stock prices due to their characteristics. R. Engle et al. (2013) proposed a mixed-data sampling component to the GARCH model, namely the GARCH-MIDAS model, which proved to be a valuable way to analyze the link between macroeconomic variables and financial volatility.

The paper of Conrad and Kleen (2019) compares the forecast performance for multiplicative volatility of stock prices of the GARCH-MIDAS model to different competitor models. They use various macroeconomic data as explanatory variables in their research. The results indicate an improvement for the forecast horizons of 2- to 3-months-ahead when using housing starts growth as explanatory variables within the GARCH-MIDAS model. Currently, interest in the use of supervised machine learning in the stock market is increasing. Therefore, it is of interest to see how the GARCH-MIDAS performs compared to LLF and to test the predictive ability of LLF on stock market volatility. Hence, this paper uses the exact data set of Conrad and Kleen (2019) in order to compare the predictive ability of the LLF of Friedberg et al. (2020) to the GARCH-MIDAS model.

There have been countless papers studying the predictive ability of GARCH-MIDAS on stock market volatility using macroeconomic data (R. Engle et al., 2013; Conrad and Loch, 2015; Fang et al., 2018). However, the research for these types of data sets on machine learning algorithms is limited. Next to this, the LLF method of Friedberg et al. (2020) has not yet been empirically tested on financial volatility. Their paper provides two other empirical results; one where they predict wages using covariates such as age, race, years of education, and gender, and the second on the relationship between word choice and public opinions on welfare. This lack of empirical research

stresses the need to explore more uses of LLF, for example, in the stock market. Higher volatility means higher risk but can also lead to higher returns. Anticipating these swings will help the trader or firm improve their strategies, for example, by buying or selling options of that asset. As a proxy for the time value of options, the expected value of the underlying asset's volatility is used. Hence, improving the forecast on long-term volatility could enhance their market position.

This paper uses the R package grf (J. Tibshirani et al., 2019) to build the LLF models, where the dependent variable is Realized Volatility (RV), and the features contain real-time macroeconomic variables. The out-of-sample forecasts are made using a rolling window, from which different loss functions are calculated (e.g., QLIKE loss, RMSE, and RMAD). Different tuning of the forests is considered, including LASSO for selecting variables and ridge penalty to prevent overfitting to the local trend. Lastly, the SHAP values are presented to indicate the importance of each variable.

The main findings suggest that the LLF method improves upon current models if the explanatory variables are chosen accordingly. The model including all macroeconomic variables performed best for all forecast horizons compared to the GARCH-MIDAS and competitor models. However, when only the lags of housing starts are included, the performance was worse than the GARCH-MIDAS model. The macroeconomic variables that proved to attribute most to the target variable are VIX, NFCI, and RVol.

The paper is organized as follows. First, a literature review on random forests and the GARCH-MIDAS is given, followed by a description of the data used. Next, the methodology is explained in detail in Section 4. Section 5 describes the results. Lastly, a conclusion is given together with a short discussion.

2   Literature review

2.1   Random forest

Random forests are nonparametric regression approaches first introduced by Breiman (2001) and builds upon recursive partitioning (CART) (Breiman et al., 1984), bagging (Breiman, 1996), and random trees (Amit and Geman, 1997). In a variety of applications, random forests have proven to be very useful (Svetnik et al. (2003); Cutler et al. (2007)). However, they suffer from significant weakness, namely that they cannot take advantage of strong local trends in estimating the regression surface. To solve this weakness, random forests can be interpreted as an adaptive kernel, following Hothorn et al. (2004), Meinshausen and Ridgeway (2006), and Athey et al. (2019), adding to the

usual understanding of forests as an ensemble approach.

It is possible to correct imbalances and asymmetries in the set of neighboring prediction points using the local regression adjustment, such that the predictions are centered and the weighted mean of the feature vector of nearby points is roughly the same as the target feature vector. When several features have a substantial impact with moderate curvature, the improvement from regression adjustments to forests is most likely to be considerable, making regression modifications both effective and useful.

There is a substantial body of work that explores various combinations of trees and local regressions. For starters, Karalic (1992) proposed to fit local linear regressions in every leaf, while Torgo (1997) emphasized the kernel performance for MOB tree models in general. Both Torgo (1997) and Gama (2004) examined functional tree leaves models, by fitting models at each node instead of local averages.

Menze et al. (2011) presented oblique random forests that use ridge regression findings to learn split directions, which is comparable to the work of Friedberg et al. (2020), establishing splitting rules for LLF but more in line with linear discriminant analysis (LDA). As introduced by Xu et al. (2016), case-specific random forests employ local information to up weight training data during the bootstrap to produce data sets for each tree, rather than during the prediction step. Zeileis et al. (2008) first, and Rusch and Zeileis (2013) later introduced recursive partitioning by fitting a different model in each leaf node, related to the residual splitting technique used in LLF. Local linear forests add to this literature; nevertheless, they vary in that they regard forests as a kernel approach. In an LLF model, the leaf nodes give neighbor information rather than local predictions.

2.1.1 Empirical studies

Recently, the use of machine learning algorithms to forecast time series has gained popularity. However, the researches are extensive, and many properties remain unexplored. Tyralis and Papacharalampous (2017) use two large data sets of high-frequency time series to evaluate the effectiveness of random forests in one-step-ahead forecasting of temperatures to achieve an optimal collection of predictor variables. They simulate time series using Autoregressive Fractionally Integrated Moving Average (ARFIMA) models together with random forests and conclude that the highest predictive performance is observed when including a low number of lagged predictor variables.

More along the lines of this paper, Luong and Dokuchaev (2018) combine a heterogeneous autoregressive (HAR) model and machine learning to forecast Realised volatility of financial time

series. The general HAR is extended by including purified implied volatility and is applied to a random forest algorithm to predict the direction and magnitude of the volatility. The data set includes high-frequency data, and their findings show improvements in the forecasting accuracy for the HAR model. Moreover, Polamuri et al. (2019) use multiple machine learning methods to attempt innovative work to predict the stock market prices. Among all the methods are Support Vector regression, linear regression, decision trees, extra tree regressor, and random forest regressor. They conclude that out of all these methods, decision trees and random forest regressors perform best.

The study of Vassallo et al. (2020) answers three questions regarding optimal random forest modeling strategies in their journey to forecast wind speed/power. The results interesting for this research focus on the effect of recursive (step-by-step) multi-step forecast versus a direct method (single jump). The direct forecasting strategy only slightly outperforms the recursive method. Next to this, the amount of training data required to benefit from asymptotic performance behavior was unattainable to compute. However, increasing the training data to approximately eight years does increase the accuracy in forecasting wind speed/power.

Lastly, Sadorsky (2021) relies on decision tree bagging and random forests to predict the stock market direction of clean energy companies. They use well-known technical indicators as features driving these prices and compare the results with logit models. Their results indicate a substantial increase in accuracy, from 55% - 60% of the logit models to 85% - 90% for tree bagging and random forest methods.

2.2 GARCH-MIDAS

Autoregressive Conditional Heteroskedasticity (ARCH) models have been used excessively to capture the time variation in volatility and the driving factors behind this, dating back to the early work of R. F. Engle (1982). This model has since been extended to the Generalized ARCH (GARCH) model by Bollerslev (1986), followed by multiple versions and extensions additive to the GARCH. At the same time, a broad literature has connected macroeconomic variables and financial instruments to volatility. Schwert (1989) has related the change in volatility of stock returns to macroeconomic variables and discusses how bond returns, short-term interest rates, producer prices, and the pace of increase in industrial output give further information on monthly market volatility. The research of Glosten et al. (1993) discovered that short-term interest rates have a significant impact on future market volatility. Whitelaw (1994) establishes statistical significance for the spread of a commercial

paper and the one-year treasury rate, whereas Brandt and Kang (2002) find statistical significance for the short-term interest rate, the term premium, and the default premium. Other studies, including Hamilton and Lin (1996) and Perez-Quiros and Timmermann (2000), have discovered evidence that the status of the economy is a significant factor of return volatility.

Since time-varying volatility analyses are usually based on high-frequency data, earlier research has been confined to variables such as short-term interest rates, term spread, and default premiums, for which daily data is available. As a result, the effects of variables like the unemployment rate and inflation have not been thoroughly investigated. Ghysels et al. (2006) introduce the mixed data sampling regression scheme (MIDAS) to allow for different frequencies to be included in the same model. This framework allows the model to combine high-frequency return data with less frequent (e.g., monthly, quarterly) macroeconomic data. R. Engle et al. (2013) propose the MIDAS structure in the GARCH model to study time-varying volatility in the stock market. Here, the conditional variances can be split into long- and short-term components affected by low- and high-frequency variables, respectively. By combining both components, the GARCH-MIDAS model is constructed. The model's key benefit is that it allows us to connect daily stock return data with macroeconomic factors collected at lower frequencies, enabling us to analyze macroeconomic variables' influence on stock volatility directly.

R. Engle et al. (2013) employ monthly inflation and industrial production rate as explanatory variables in their study to model stock market volatility with the GARCH-MIDAS component model. Their findings suggest an improvement in terms of long-horizon forecasting compared to more traditional time series volatility models. Conrad and Loch (2015) investigate the relation between quarterly macroeconomic factors and long-term US stock market volatility using a two-component GARCH-MIDAS model. Their findings indicate that housing starts, the term spread, unemployment rate, and corporate profits exceed the predictive ability for the long-term component. Additionally, Asgharian et al. (2013) use principal component analysis combined with GARCH-MIDAS to include information enclosed in various variables. The results suggest that including low-frequency macroeconomic data into the GARCH-MIDAS model enhances the model's prediction performance, especially for the long-term variance component. Furthermore, the GARCH-MIDAS model with the first principal component surpasses all other specifications, implying that the created principal component is a reliable proxy for the business cycle. Opschoor et al. (2014) use a Spline-GARCH type model, which can be considered a special case of the GARCH- MIDAS, where only the first lag of the explanatory variable X is used in the specification of the long-term component. Their

empirical research considers daily US stock returns of deposit banks and the Financial Conditions Index (FCI) by Bloomberg as a proxy for the financial state of bond, equity, and money markets. They find an improvement of the Value-at-Risk estimation by including the FCI, particularly for shorter horizons. A further application of the GARCH-MIDAS model has been proposed by Dorion (2016), where he allows for changes in business cycles for valuation of options, referring to it as the MacroGARCH model. Next to predicting the stock market volatility, the GARCH-MIDAS model can also be used to forecast the long- and short-term components of cryptocurrency volatility (Conrad et al., 2018). Measures of risk and volatility in the US stock market and global economic activity measures are considered as potential drivers of the volatility of Bitcoin. Atypically, their findings suggest a negative effect of stock market volatility on long-term Bitcoin volatility. They also show that there is a strong link between global economic activity and Bitcoin volatility.

More recently, Conrad and Loch (2015), Conrad and Kleen (2019), and Wu and Chong (2021) incorporate a more broad set of macroeconomic factors in the GARCH-MIDAS model. They all draw similar conclusions where the most influential variables are term spread, housing starts, the change in unemployment rate, the default rate, and the National Activity Index (NAI). The predictive performance of GARCH-MIDAS is compared to the Realized GARCH of Hansen et al. (2012), the Markov-switching GARCH, the heterogeneous autoregression (HAR) model of Corsi (2009), and the high-frequency-based volatility (HEAVY) model of Shephard and Sheppard (2010). Specifically, the results of Conrad and Kleen (2019) indicate that the GARCH-MIDAS approach with housing starts growth as the explanatory variable performs better than competitor models for 2- and 3-months-ahead forecasting horizons.

3   Data

All data has been retrieved from the Journal of Applied Econometrics data archive, following the data from Conrad and Kleen (2019). The data set consists of the returns of the S&P 500 and several macroeconomic variables, including their Real-time release dates.

The stock market return data consists of the daily log return of the S&P 500 for the period from January 1971 to April 2018. The daily realized variance ($RV_{i,t}$) has been defined by Conrad and Kleen (2019) based on the five-minute intraday returns and are available from 2000 onwards. The data to construct these are provided by the Realized Library of the Oxford-Man Institute of Quantitative Finance.

The explanatory variables which are used in the model are summarized below; a more elaborate description of the data is presented in Table 6 of Appendix A.

- **NFCI**: As a daily measure of financial risk, the Chicago Fed's National Financial Conditions Index (NFCI) is used. Positive values of the NFCI indicate tighter financial conditions than average, while the opposite is true for negative values.

- **NAI**: The Chicago Fed National Activity Index (NAI) is a monthly measure of macroeconomic variables. It measures the weighted mean of 85 economic variables. The NAI takes values that are positive/negative whenever the US economy is expanding/declining.

- **ΔIP**: Another monthly macroeconomic variable is the Industrial Production Index (IP) which measures the changes in Real output. The explanatory variable consists of the growth rates for IP calculated as the log differences.

- **ΔHous**: The growth rates of housing starts calculated as the log differences of the New Privately Owned Housing Units Started (HOUST).

- **RVOL**: As a backward measure of daily volatility, the average realized volatility over a rolling window of 22 days is used, $RVOL(22)_{i,t} = \sqrt{1/22 \sum_{j=0}^{21} r_{i-j,t}^2}$.

- **VIX**: For the forward measure of daily volatility, the VIX index is used. This index provides the expected volatility of the S&P 500 and can be retrieved from the Chicago Board Options Exchange. The VIX is divided by $\sqrt{252}$ in order to convert it to a daily level.

- **VRP**: The difference between RVOL22 and the square root of the VIX is used as a proxy for the variance risk premium (VRP).

The choice of lag length for the explanatory variables differs across the measures, as the frequency between these variables variates. Conrad and Kleen (2019) use a lag length of 3 for the VIX, a length of 52 for the NFCI and a length of 36 for the monthly macroeconomic variables, NAI, ΔIP and ΔHous. Some will be included as covariates in the models, more on that in Section 4.

A Real-time database is included in the data set, which is a collection of all vintages from the same time series, indexed by the date each data point was released to the public. The latest revision for each reference point at a particular moment in time is used. Conrad and Kleen (2019) emphasize the need to use Real-time data rather than the final releases of macroeconomic data for a

9

representational forecast evaluation. A "look-ahead bias" is eliminated by utilizing Real-time data since macroeconomic indicators are revised significantly after their first publication.

4 Methodology

The methodology is subdivided into five sections. Section 4.1 contains a description of the LLF model. The second section explains the GARCH-MIDAS model. The third and fourth sections address forecasting procedures and measures needed to compare the forecasting performances respectfully. Lastly, the fifth section explains the SHAP values, which are used to see which macroeconomic variable contributes the most to the prediction of the realized variance. A description of the code implementing all the methodology is mentioned in Appendix D.

4.1 Local Linear Forest

Local linear regression can be estimated using weights as a kernel which can be provided by random forest. Assuming we have (X_1, Y_1), ..., (X_n, Y_n) available training data with $Y_i = \mu(X_i) + \epsilon_i$ for i = 1, ..., n, the conditional mean $\mu(x_0) = \mathbb{E}[Y|X = x_0]$ can be estimated using random forest, where x_0 is a fixed test point. For the use of this paper, the value of Y_i is defined as the $RV_{i,t}$ of the S&P 500. Next to this, the value of X_i variates across models in order to achieve maximum forecast accuracy. Random forests are traditionally considered ensemble methods, which are defined to create multiple learning algorithms to improve predictive performance (Opitz and Maclin, 1999). In the case of random forests, the final estimate is obtained by averaging the predictions of the trees. Each leaf $L_b(x_0)$ from tree T_b has a predicted response $\hat{\mu}_b(x_0)$, which is the average overall training data assigned to leaf $L_b(x_0)$. From these responses, the average prediction over all trees is $\hat{\mu}(x_0) = (1/B) \sum_{b=1}^{B} \hat{\mu}_b(x_0)$ where B is the number of trees in the forest.

Alternatively, the average prediction $\hat{\mu}(x_0)$ can be written as the weighted sum of all the dependent variables in the training set Y_i. Here, random forests are viewed as adaptive weight generators (Hothorn et al., 2004; Meinshausen and Ridgeway, 2006; Athey et al., 2019). The predicted conditional mean can be written as

$$\hat{\mu}(x_0) = \sum_{i=1}^{n} \alpha_i(x_0) Y_i, \tag{1}$$

where $\alpha_i(x_0)$ are the forest weights defined as

$$\alpha_i(x_0) = \frac{1}{B} \sum_{b=1}^{B} \frac{I\{X_i \in L_b(x_0)\}}{|L_b(x_0)|} \quad \text{s.t.} \quad 0 \leq \alpha_i(x_0) \leq 1. \tag{2}$$

Given that there exists at least one non-empty cell containing x_0, $\sum_{i=1}^{n} \alpha_i(x_0) = 1$. LLF take $\alpha_i(x_0)$ and use them in order to predict the local linear regression:

$$\begin{pmatrix} \hat{\mu}(x_0) \\ \hat{\theta}(x_0) \end{pmatrix} = \text{argmin}_{\mu,\theta} \left\{ \sum_{i=1}^{n} \alpha_i(x_0)(Y_i - \mu(x_0) - (X_i - x_0)\theta(x_0))^2 + \lambda \|\theta(x_0)\|_2^2 \right\}, \tag{3}$$

where the slope parameter $\theta(x_0)$ corrects the local trend in $X_i - x$ and the term $\lambda\|\theta(x_0)\|_2^2$ is added as a ridge penalty to prevent overfitting to the local trend. It plays an important role in simulation experiments as well as asymptotic convergence results. The weighted least squares Equation 3 with weights (2) is solved by LLF. This estimator can then be written as

$$\begin{pmatrix} \hat{\mu}(x_0) \\ \hat{\theta}(x_0) \end{pmatrix} = \left(\Delta^T A \Delta + \lambda J\right)^{-1} \Delta^T A Y. \tag{4}$$

Throughout this paper, the weight matrix A is a diagonal matrix with $A_{i,i} = \alpha_i(x_0)$ and J is an identity matrix with $J_{1,1} = 0$ such that the intercept is not penalized. The centered regression matrix, Δ, is defined as $\Delta_{i,1} = 1$ and $\Delta_{i,j+1} = x_{i,j} - x_{0,j}$.

The local linear regression can be thought of as a function with weights $\gamma_i \alpha_i(x_0)$, defining $\gamma_i = e_i \left(\Delta^T A \Delta + \lambda J\right)^{-1} \Delta^T$ with e_i being a zeros vector with value 1 at column i. Now, the moments of x_0 are better oriented around the test point: $\hat{\mu}(x_0) = \sum_{i=1}^{n} \gamma_i \alpha_i(x_0) Y_i$ with $\sum_{i=1}^{n} \gamma_i \alpha_i(x_0) = 1$ and $\sum_{i=1}^{n} \gamma_i \alpha_i(x_0) X_i \approx x_0$.

4.1.1 Splitting rule

Classification and regressions tree (CART) splits, introduced by Breiman et al. (1984), are traditionally used by random forests. The standard CART splits determine the splitting value by minimizing the sum of squared errors (SSE) for every possible variable. First the parent node P is considered with n_p observations. The mean of Y for all child pair nodes C_1, C_2 gives $\bar{Y}_1$ and $\bar{Y}_2$ respectively. After this, C_1 and C_2 are chosen based on the minimum SSE:

$$SSE = \sum_{i:X_i \in C_1} (Y_i - \bar{Y}_1)^2 + \sum_{i:X_i \in C_2} (Y_i - \bar{Y}_2)^2. \tag{5}$$

However, there is no need for the forest to model the strong and smooth signal as the last

regression step models them. Instead, ridge regression in node P can be used to predict Y_i:

$$\hat{Y}_i = \hat{\alpha}_P + x_i^T \hat{\beta}_P, \tag{6}$$

where $\hat{\alpha}_P$ is the intercept and $\hat{\beta}_P = (x_P^T x_P + \lambda J)^{-1} x_P^T Y_P$. Standard CART splits can now be used on $Y_i - \hat{Y}_i$, such that the local effects are modeled in the forest, and the global effects are regressed back in when predicting. This splitting rule can be enforced on every variable giving balanced splits necessary for honest forests.

For tuning the random forests in this paper, the ridge penalty for splitting is set to $\lambda = 0.10$, and the ridge penalty for predicting is tuned by default using the R package grf (J. Tibshirani et al., 2019). This feature selects the penalties by cross-validation.

4.1.2 Honest Forests

There are different ways to grow trees in a random forest. Friedberg et al. (2020) mentioned that there are no results available on the consistency of random forests where the trees are fully grown for cases without the use of honesty. Hence, honest forest is used in the entirety of this paper, which is a way of sub-sample splitting. Wager and Athey (2018) used honest forest to derive the asymptotic properties of prediction using random forests. Two independent subsamples are drawn from the training data, I_b and J_b. First, the tree structure T_b is chosen only using J_b and let $x_0 \Longleftrightarrow_b x'$ be a boolean indicating whether x_0 and x' end up in the same leaf node of T_b. Secondly, let $L_b(x_0) = \{i \in I_b : x_0 \Longleftrightarrow_b X_i\}$ be the set of neighbors of x_0. This neighborhood function is used to define the weights in Equation (2). The y-values from sample I_b are not used to determine split points. The covariates, however, may be used to ensure that a fraction from the parent is incorporated in each node. Using this modification, the forest complies with the assumption that trees are split on every variable with probability larger than zero, symmetric in permutations, and balance the observations of the parent in every child node.

4.1.3 Forest size

Selecting the appropriate amount of trees in a forest is important as the computational costs increase together with the number of trees. Studies have suggested a wide range in the number of trees. For example, the study of Oshiro et al. (2012) suggests a number between 64 and 128 trees, while Kuhn and Johnson (2013) suggest incorporating a minimum of 1000 trees. On the other hand, Probst

and Boulesteix (2017) suggest training 100 trees to achieve maximum performance gain. For this study, the number of trees is set to 500.

4.1.4 Feature selection

In order to achieve the best forecast, multiple models are considered and compared. Only the covariates differ across the models, with the dependent variable being Realized volatility. The selection of variables is partly based on the succeeding models used in Conrad and Kleen (2019): the first model only contains Δ Hous with lag up to 36. The second model contains all variables mentioned in Section 3 without their lags. Finally, the last model contains the same variables as the second model, with the lag of the log returns added to the set. On these data sets, two different approaches are examined to specify the number of covariates on which the splits are performed: one where LASSO is used to choose the covariates that are incorporated in the model and one where all the covariates are used.

LASSO (least absolute shrinkage and selection operator) was introduced by R. Tibshirani (1996), and it results in stable and interpretable models. LASSO performs variable selection and regularization to enhance the predictive accuracy of the model. It minimized the sum of squares conditional on the sum of the absolute values of all coefficients being less than a constant. This method is also included in the R package grf (J. Tibshirani et al., 2019) and is run beforehand on the training data set. The returned covariates are then implemented in the LLF prediction method; the results can be found in Table 1 in the Results.

4.2 GARCH-MIDAS

The daily log returns can be modeled as $r_{i,t} = \mu + \epsilon_{i,t}$ where $t = 1, ..., T$ indicates a period (week or month) and $i = 1, ..., I_t$ are the number of days within that period. The GARCH-MIDAS model of R. Engle et al. (2013) differs from the standard GARCH model because it allows the components to vary in frequencies. The demeaned scaled returns can be noted as:

$$\frac{\epsilon_{i,t}}{\sqrt{\tau_t}} = \sqrt{g_{i,t}} Z_{i,t}, \tag{7}$$

where τ_i is the long-term volatility component and is a function of explanatory variable X_t. The short-term volatility component, $g_{i,t}$, follows from the GARCH model, and $Z_{i,t}$ are i.i.d. with zero mean and variance equal to one. The information set of day i in period t is $F_{i,t}$ with $F_t := F_{I_t t}$.

The conditional variance of the returns can be defined as

$$\sigma_{i,t}^2 := g_{i,t}\tau_t, \tag{8}$$

if τ_t only depends on the lagged values of X_t; equivalently, $\sigma_{i,t}^2 = \text{var}(\epsilon_{i,t}|F_{i-1,t})$. The short-term variance component is constructed such that it describes day-to-day volatility clustering and is presumed to follow a GJR-GARCH(1,1) process:

$$g_{i,t} = (1 - \alpha - \gamma/2 - \beta) + (\alpha + \gamma \mathbb{1}_{\{\epsilon_{i-1,t}<0\}})\frac{\epsilon_{i-1,t}^2}{\tau_t} + \beta g_{i-1,t}. \tag{9}$$

Here, the convention $\epsilon_{0,t} = \epsilon_{I_{t-1},t-1}$ and $g_{0,t} = g_{I_{t-1},t-1}$ is used, as well as $\tau_{i,t} = \tau_t$ for $i = 1, ..., n$ and $\tau_{I_{t-1},t-1} = \tau_{t-1}$. If $I_t > 1$, then τ_t is a constant. For $I_t = 1$, both the short- and long-term component vary at equal frequencies, such that index i can be dropped.

Some assumptions are made about the innovation process $Z_{i,t}$ and short-term component parameters.

Assumption 1. $Z_{i,t}$ are i.i.d. random variables where $\mathbf{E}[Z_{i,t}] = 0$, $\mathbf{E}[Z_{i,t}^2] = 1$, and $k = \mathbf{E}[Z_{i,t}^4]$ for $1 < k < \infty$.

Assumption 2. Let $\alpha > 0$, $\alpha + \gamma > 0$, $\beta \geq 0$, $\alpha + \gamma/2 + \beta < 1$, and $(\alpha + \gamma/2)^2 k + 2(\alpha + \gamma/2)\beta + \beta^2 < 1$.

Taken both assumptions together, the scaled returns defined in Equation 7 follow a covariance stationary process. The first-order moment of $g_{i,t}$ is equal to $\mathbf{E}[g_{i,t}] = 1$, the second-order is defined as

$$\mathbf{E}[g_{i,t}^2] = \frac{1 - (\alpha + \gamma/2 + \beta)^2}{1 - (\alpha + \gamma/2)^2 k - 2(\alpha + \gamma/2)\beta - \beta^2}, \tag{10}$$

and lastly, the fourth moment is finite. The long-term component, τ_t, has the role of describing smooth movements within the conditional variance. This component is generally specified as a positive-valued, measurable function $f(\cdot)$ of its explanatory variable X_t together with the lags $K \geq 1$:

$$\tau_t = f(X_t, X_{t-1}, ..., X_{t-K}). \tag{11}$$

An additional assumption is made about the specifications of this function $f(\cdot)$ and X_t.

Assumption 3. Function $f(\cdot) > 0$ is measurable and X_t is an ergodic, strictly stationary time

series where $\mathbf{E}[|X_t|^n] < \infty$ with n large enough such that $\mathbf{E}[\tau_t^2] < \infty$. $Z_{i,t-j}$ and X_t are independent $\forall t, i$ and j.

This assumption implies strict stationary, covariance stationary and does not depend on $g_{i,t}$ for τ_t. The choice for function $f(\cdot) > 0$ is generally a simple linear model:

$$\tau_t = a + \pi_1 X_t - 1 + ... + \pi_K X_t - K, \tag{12}$$

with the requirements that $a > 0$ and $\pi_m \geq 0$ for $m = 1, ..., K$ and X_t is nonnegative. However, if X_t can take on negative values the exponential of Equation (12) can be taken. The function for τ_t now becomes:

$$\tau_t = \exp(a + \pi_1 X_t - 1 + ... + \pi_K X_t - K). \tag{13}$$

The weights π_m are determined by a weighting scheme, where the most common choice is $\pi_m = \theta \cdot \phi_m(w_1, w_2)$. The parameter θ regulates the sign of the effect X_t has on τ_t, and the weights can be estimated using the following Beta weighting scheme:

$$\phi_m(w_1, w_2) = \frac{[m/(K+1)]^{w_1-1} * [1 - m/(K+1)]^{w_2-1}}{\sum_{j=1}^{K}[j/(K+1)]^{w_1-1} * [1 - j/(K+1)]^{w_2-1}}. \tag{14}$$

The sum of the weights are equal to 1; that is, $\sum_{m=1}^{K} \phi_m(w_1, w_2) = 1$. It follows that $\mathbf{E}[\tau_{t+1}|F_t] = \tau_{t+1}$. A note about the independence assumption of X_t and $Z_{i,t-j}$, as it might appear to be too strong: even if there would be dependence between daily innovations and the daily X_t, the dependence between $Z_{i,t-j}$ and τ_t is probable to be negligible due to the smoothness of the function of τ_t. Conrad and Kleen (2019) provide an illustration through simulations that light independence violations do not affect the main results.

4.3 Forecasting

The forecast period ranges from 2010M1 to 2018M1, with forecasts horizons of cumulative volatility up to three months. Out-of-sample forecast evaluations are made using a rolling window for all models. The rolling window length depends on the sample size of the data set for each model, starting from the first available point in the data set up to and including 2009M12. Next to the full out-of-sample forecasts, three subsamples are considered, each representing a different volatility regime: namely, low, normal, and high. Finally, a distribution is made of the daily realized variances

from the complete empirical data set. If a forecast is below the threshold of the 25% quantile, it falls in the low regime. Similarly, the forecast falls in the normal or high regime if it is in between the quantile of 25% and 75% or above the 75% quantile threshold.

The realized variances used for evaluation are defined as the daily realized variance adjusted for the squared difference in return and open-close price. The cumulative sum for each proxy can now be written as: $RV_{1:k,t+s}$, where k is defined as the number of forecasting days in each month and s states the number of months. The realized variances are forecasted using multiple-day ahead horizons, namely one day (1d), two weeks (2w), one month (1m), two months (2m), or three months (3m), i.e. $k \in \{1, 10, 22\}$ and $s \in \{1, 2, 3\}$ respectively. For the forecast horizons of more than one day, the forest is held constant for consecutive dates until the horizon. For example, 2-months-ahead forecasts at time $t + 2$ are estimated by training a forest using data from $t - 2$ to t, where s is the length of the forecast window. The covariates from time t to $t + 2$ are then plugged into the obtained forest to make the 2-months-ahead forecast. This approach is also implemented by Medeiros et al. (2021).

4.4 Performance evaluation

The forecasts are compared based on their QLIKE loss, and after this, a model confidence set (MCS), introduced by Hansen et al. (2011), is used to compare these losses. In addition, the different models are compared based on a range of loss functions.

Firstly, the QLIKE loss function is the single robust function which depends only on $\frac{\sigma^2_{k,t+1}}{h_{k,t+1|t}}$ (standardized forecast error). For k-step-ahead forecasts, the function can be defined as:

$$\text{QLIKE}(\sigma^2_{k,t+1}, h_{k,t+1|t}) = \frac{\sigma^2_{k,t+1}}{h_{k,t+1|t}} - \ln(\frac{\sigma^2_{k,t+1}}{h_{k,t+1|t}}) - 1. \tag{15}$$

Here, $\sigma^2_{k,t+1}$ represents the realized variance and $h_{k,t+1|t}$ is the forecasted variance. The loss functions are jointly evaluated through MCS. The objective of this procedure is to return the best models, M^*, from a collection of all models M^0. The difference between the QLIKE functions i and j is defined as:

$$d_{i,j}(s, k) = \text{QLIKE}(RV_{1:k,t+s}, \hat{h}^{(i)}_{1:k,t+s|t}) - \text{QLIKE}(RV_{1:k,t+s}, \hat{h}^{(j)}_{1:k,t+s|t}). \tag{16}$$

In Equation (16), $\sigma^2_{k,t+1}$ is replaced by $RV_{1:k,t+s}$ such that the long-term component of Equation (7) is fixed. Here, $k \in \{1, 10, 22\}$ denotes the daily components (short-run) and $s \in \{1, 2, 3\}$ denotes

the monthly component (long-run).

Then the test statistic can be calculated:

$$t_{ij} = \bar{d}_{i,j}/\sqrt{\hat{\text{var}}(\bar{d}_{i,j})} \qquad \forall i, j \in M^0. \tag{17}$$

The test statistic of MCS follows from $T_M = \max\limits_{i,j \in M} |t_{i,j}|$ with H_0: the expected loss of all models is the same. Under the alternative hypothesis, there exists a model i whose expected loss is greater than the rest of the models $j \in M/i$. If this is the case, this model is eliminated from the rest, and the test is performed again until there are no models left to be eliminated. The surviving models are denoted by M_{MCS} and are defined as the best forecast models with confidence $1 - \nu$. The choice of ν is commonly set to 0.1, as is done by Laurent et al. (2013) and Liu et al. (2015).

The different loss functions include the Root Mean Squared Error (RMSE), the Root Mean Absolute Error (RMAE), the Root Mean Standard Deviation (RMSD), and the Root Mean Absolute Deviation (RMAD). Their formulas are as follows:

$$RMSE = \sqrt{\frac{1}{T} \sum_{i=1}^{T} (\sigma_{i+1}^2 - E(\sigma_{i+1}^2))^2}, \tag{18}$$

$$RMAE = \sqrt{\frac{1}{T} \sum_{i=1}^{T} |\sigma_{i+1}^2 - E(\sigma_{i+1}^2)|}, \tag{19}$$

$$RMSD = \sqrt{\frac{1}{T} \sum_{i=1}^{T} (\sigma_{i+1} - E(\sigma_{i+1}))^2}, \tag{20}$$

$$RMAD = \sqrt{\frac{1}{T} \sum_{i=1}^{T} |\sigma_{i+1} - E(\sigma_{i+1})|}, \tag{21}$$

where σ_{i+1}^2 is the actual realized variance and $\mathbf{E}(\sigma_{i+1}^2)^2$ is the forecasted variance, T denotes the number of forecasted values.

4.5 SHAP values

As mentioned earlier, it is known that certain macroeconomic variables have an impact on the variance of stock prices. In order to find out which macroeconomic variable contributes most to the prediction of Realized variances, the SHapley Additive exPlanations (SHAP) are used. It was first

developed by Lundberg and Lee (2017) and helps to gain more knowledge on the so-called 'black box' which are machine learning models.

The SHAP method assigns a value of importance to each feature for a certain prediction. This is done using different subsets of the entire variable set: $S \subseteq F$. Two models are trained, one where all features are included in the trained model f_S, and one where only feature i is included $f_{S \cup \{i\}}$. Both predictions are compared to each other $f_{S \cup \{i\}}(x_{S \cup \{i\}}) - f_S(x_S)$, where x_S are the values of the features in subset S. This process continues for all possible subsets, $S \subseteq F \setminus \{i\}$. Consider the prediction model f with explanation model g. The latter model uses *simplified input variables* x', which map back to x using a mapping function $x = h_x(x')$. The SHAP values differ from Shapley values because they use a conditional expectation model retrieved from the original model as mapping function.The explanation function g is a linear function using binary inputs z as explanatory variables:

$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z_i', \tag{22}$$

where M is the total number of inputs and $z' \in \{0,1\}^M$. The effect to each feature is described by ϕ_i, which sum approximates the original output $f(x)$. The SHAP values can then be defined as:

$$\phi_i(f, x) = \sum_{z' \subseteq x'} \frac{|z'|!(M - |z'| - 1)!}{M!} [f_x(z') - f_x(z' \setminus i)]. \tag{23}$$

Here, $f_x(z') = f(h_x(z')) = E(f(z)|z_S)$ with S being the set where $z_i' = 1$. The initial input variables are all variables mentioned in Section 3 without their lags, but including the lag return. The dependent variable is the realized variance. As the LLF model is relatively new, there has not yet been a development on the computation of SHAP values using LLF as the predictor model. Hence, an Extreme Gradient Boosting (XGBoost) model is used. This model is an ensemble machine learning algorithm using decision trees. The objective of using the SHAP is to see which features are important in predicting the realized variances, hence using another machine learning algorithm still gives a relevant insight. The code of Lundberg (2019) is used to estimate the SHAP (package shap).

5 Results

This section discusses the results of all the models. In the paper of Conrad and Kleen (2019), the GARCH-MIDAS model was compared to various competitor models. Some of them outperformed the GARCH-MIDAS model in terms of forecasting accuracy, which is why these methods are included in the comparison to LLF. These models include the Realized GARCH (Hansen et al., 2012), the heterogeneous autoregression (HAR) model (Corsi, 2009), and the HAR with leverage effect (Corsi and Renò, 2012).

First, the estimation results of the relevant GARCH-MIDAS models are presented in Appendix B, where Table 7 refers to the GARCH-MIDAS model with one covariate and Table 8 refers to the GARCH-MIDAS model with covariate VIX in combination with another variable. Second, the LASSO results for every LLF model are presented below in Table 1. As is evident from this table, LASSO is not applicable for the model only considering the lags of Δ Hous. Hence, there will be no results from LASSO on that model in this section. For the model including all variables, the selected variables from LASSO are RVol(22) and VIX. LASSO selects the same variables for the model when the lag of the return is included.

Table 1

LASSO results for different LLF models

	RVol (22)	VIX	VRP	NFCI	NAI	Δ IP	Δ Hous	$r_{i,t-1}$
Dep. var. $RV_{i,t}$								
Δ Hous (36)	-	-	-	-	-	-	no	-
All var	yes	yes	no	no	no	no	no	-
All var + $r_{i,t-1}$	yes	yes	no	no	no	no	no	no

Note. These are the LASSO results performed on training data set up to and including 2009M12. Yes/no are indicators whether LASSO includes the variables.

As mentioned in Section 4.4, the forecast performance of various models is compared based on the QLIKE losses. The results of these losses on the complete data set are presented in Table 2 below. The shaded entries imply the surviving models in the set c with a 90% confidence interval. For example, the models included in M_{MCS} for the 1-day-ahead forecast horizon are all LLF models except the one including only the lags of Δ Hous. For the 2-weeks-ahead horizon, the models in M_{MCS} are only the LLF models with all variables with and without the return variable, thus excluding the LASSO models. The LLF for all variables with and without the lag of the return, not including LASSO, stays in M_{MCS} for all horizons. For 2-months- and 3-months-ahead, the GARCH-MIDAS using the lags of housing starts improves upon the competitor models but still

have significantly higher losses than the LLF.

The bold entries indicate the lowest average loss for that particular forecast horizon. The data in the table shows that the LLF models using all variables (in combination with LASSO and $r_{i,t-1}$) attain the lowest losses and are thus dominant over the other models in terms of predictive ability. Both the model with all variables as all variables with the lag return are in M_{MCS} for each horizon, and they score the lowest values for 2-weeks-, 1-month-, 2-months- and 3-months-ahead. By comparing the numbers from these models with GARCH-MIDAS, it is clear that LLF proves to be very useful provided the correct variables are used to train the model.

Table 2

QLIKE losses and MCS results per model on the complete data set

		1d	**2w**	**1m**	**2m**	**3m**
LLF	Δ Hous	0.805	0.597	0.567	0.594	0.632
	All var	0.205	0.054	**0.037**	**0.039**	**0.040**
	All var - LASSO	**0.203**	0.056	0.039	0.039	0.041
	All var + $r_{i,t-1}$	0.206	**0.054**	0.037	0.039	0.040
	All var + $r_{i,t-1}$ - LASSO	0.204	0.056	0.039	0.039	0.041
GARCH-MIDAS	VIX	0.275	0.217	0.242	0.359	0.413
	Δ Hous	0.328	0.254	0.265	0.347	0.380
	VIX and NFCI	0.274	0.216	0.238	0.349	0.400
	VIX and NAI	0.275	0.217	0.242	0.358	0.409
	VIX and Δ IP	0.274	0.216	0.241	0.355	0.409
	VIX and Δ Hous	0.280	0.220	0.244	0.351	0.410
Competitors	HAR	0.254	0.212	0.244	0.368	0.419
	HAR (leverage)	0.238	0.209	0.246	0.371	0.420
	Real GARCH	0.245	0.206	0.234	0.356	0.369

Note. The numbers represent the average QLIKE losses of 1-day-, 2-weeks-, 1-month-, 2-months- and 3-months-ahead forecasts. The bold entries show the lowest value of average QLIKE loss across the models for that particular forecast horizon. The shaded entries show which models are in the 90% model confidence set.

Additional to the complete data set, it is interesting to consider the results for subsamples of different volatility regimes. Hence, three quantiles are considered: low, normal, and high. These regimes are computed as defined in Section 4.3 using the computed empirical distribution of the realized variances. The analysis of each regime is presented in Table 3. Here, approximately the same results are obtained as in Table 2. The LLF models, except the one with only housing starts, perform the best for almost all horizons and every volatility regime. Next to this, both HAR models and the Real GARCH are included in M_{MCS} for the one-day-ahead forecasts in the low regime. Only the HAR with leverage in combination with the GARCH-MIDAS with variables VIX + NFCI and

VIX + Δ IP shows up in the M_{MCS} for the one-day-ahead forecasts in the high regime additional to the aforementioned LLF models. The values in these columns are higher as well. Overall the values for the normal regime are for each horizon lower, indicating that the LLF models perform best under normal circumstances.

Table 3

QLIKE losses and MCS results for different volatility regimes

		low-volatility regime					normal-volatility regime					high-volatility regime				
		1d	2w	1m	2m	3m	1d	2w	1m	2m	3m	1d	2w	1m	2m	3m
LLF	Δ Hous	0.960	0.773	0.727	0.689	0.687	0.544	0.441	0.439	0.542	0.575	1.243	0.643	0.567	0.521	0.675
	All var	0.215	0.055	**0.039**	0.040	0.043	0.189	**0.053**	**0.037**	**0.041**	**0.042**	**0.229**	0.054	**0.032**	0.028	0.028
	All var - LASSO	0.213	0.056	0.040	0.040	0.043	**0.187**	0.056	0.039	0.043	0.044	0.232	0.058	0.034	0.028	0.027
	All var $+ r_{i,t-1}$	0.216	**0.054**	0.039	0.040	**0.043**	0.189	0.054	0.037	0.042	0.042	0.233	**0.054**	0.033	0.027	**0.027**
	All var $+ r_{i,t-1}$ - LASSO	**0.213**	0.055	0.040	**0.039**	0.043	0.188	0.056	0.039	0.043	0.044	0.234	0.057	0.034	**0.027**	0.027
GARCH-MIDAS	VIX	0.332	0.210	0.250	0.367	0.354	0.233	0.209	0.234	0.355	0.454	0.261	0.259	0.243	0.347	0.437
	Δ Hous	0.402	0.277	0.300	0.386	0.406	0.279	0.238	0.251	0.331	0.385	0.295	0.249	0.219	0.300	0.298
	VIX and NFCI	0.331	0.212	0.250	0.364	0.351	0.235	0.208	0.231	0.345	0.440	0.254	0.249	0.229	0.321	0.391
	VIX and NAI	0.333	0.212	0.254	0.367	0.354	0.234	0.209	0.234	0.358	0.449	0.259	0.254	0.237	0.334	0.417
	VIX and Δ IP	0.332	0.211	0.251	0.363	0.351	0.234	0.208	0.233	0.354	0.450	0.257	0.253	0.237	0.338	0.424
	VIX and Δ Hous	0.330	0.215	0.254	0.364	0.357	0.237	0.210	0.236	0.358	0.448	0.265	0.260	0.245	0.348	0.425
Competitors	HAR	0.234	0.189	0.254	0.359	0.385	0.243	0.216	0.243	0.374	0.430	0.340	0.257	0.230	0.371	0.470
	HAR (leverage)	0.226	0.187	0.258	0.362	0.387	0.232	0.215	0.241	0.378	0.429	0.286	0.245	0.227	0.373	0.470
	Real GARCH	0.225	0.197	0.260	0.380	0.372	0.256	0.200	0.222	0.343	0.401	0.323	0.247	0.209	0.260	0.258

Note. The numbers represent the average QLIKE losses of 1-day-, 2-weeks-, 1-month-, 2-months- and 3-months-ahead forecasts for volatility regimes low/normal/high; the low regime is defined as volatility in the lower 25% quantile of the empirical data set, the normal regime is in between 25% and 75%, and high regime above 75% quantile. he bold entries show the lowest value of average QLIKE loss across the models for that particular forecast horizon. The shaded entries show which models are in the 90% model confidence set.

22

Tables 4 and 5 show the mean prediction errors of the different models per horizon. For the RMSE, letting LASSO select the covariates used in prediction seems to improve the model slightly. The closer the values are to zero, the better the fit of the model to the data. The addition of the lag return does not seem to affect the errors significantly, as these outcomes stay relatively the same for all horizons. The LLF model including 36 lags of housing starts performs the worst out of all five LLF models for every loss function. By comparing these results with the GARCH-MIDAS and competitor models, the conclusion drawn from the QLIKE losses can be confirmed. The LLF models, excluding Δ Hous, outperform all GARCH-MIDAS and competitor models.

Table 4

Out-of-sample root mean squared prediction errors for all forecast horizons

RMSE		1d	2w	1m	2m	3m
LLF	Δ Hous	0.954	1.128	1.123	1.060	1.322
	All var	0.365	0.367	0.372	0.364	0.369
	All var - LASSO	0.358	**0.361**	**0.364**	0.360	**0.364**
	All var + $r_{i,t-1}$	0.364	0.365	0.374	0.363	0.368
	All var + $r_{i,t-1}$ - LASSO	**0.357**	0.362	0.365	**0.359**	0.365
GARCH-MIDAS	VIX	0.490	0.616	0.632	0.672	0.697
	Δ Hous	0.496	0.619	0.664	0.704	0.707
	VIX and NFCI	0.469	0.591	0.610	0.645	0.663
	VIX and NAI	0.484	0.608	0.624	0.663	0.687
	VIX and Δ IP	0.482	0.608	0.625	0.665	0.688
	VIX and Δ Hous	0.493	0.620	0.639	0.677	0.698
Competitors	HAR	0.404	0.563	0.636	0.753	0.820
	HAR (leverage)	0.388	0.567	0.641	0.754	0.819
	Real GARCH	0.413	0.527	0.565	0.618	0.642

Note. The table shows the loss values of the root mean squared error function for the prediction of the realized variance for all forecast horizons. The out-of-sample forecasts cover 2010M1-2018M1.

Second is the RMAE, which values are given in Table 5. This function has the advantage that every error has the same impact on the RMAE because it is based on absolute error, contrary to the RMSE. For all horizons except one month, the lowest values are obtained for the LLF model with all variables including the lag return and with the use of LASSO. The other LLF models including all variables, produce similar results but clearly outperform the other models. Again, this confirms the previous results.

Table 5

Out-of-sample root mean absolute prediction errors for all forecast horizons

| RMAE		1d	2w	1m	2m	3m
LLF	Δ Hous	0.857	0.944	0.935	0.915	0.982
	All var	0.478	0.482	0.482	0.482	0.483
	All var - LASSO	0.475	0.480	**0.479**	0.479	0.482
	All var + $r_{i,t-1}$	0.478	0.481	0.484	0.480	0.483
	All var + $r_{i,t-1}$ - LASSO	**0.474**	**0.479**	0.481	**0.478**	**0.481**
GARCH-MIDAS	VIX	0.570	0.644	0.666	0.690	0.704
	Δ Hous	0.593	0.676	0.714	0.744	0.750
	VIX and NFCI	0.564	0.637	0.660	0.681	0.692
	VIX and NAI	0.568	0.643	0.664	0.688	0.701
	VIX and Δ IP	0.567	0.642	0.663	0.687	0.701
	VIX and Δ Hous	0.573	0.650	0.673	0.697	0.710
Competitors	HAR	0.502	0.623	0.680	0.754	0.797
	HAR (leverage)	0.490	0.623	0.682	0.753	0.798
	Real GARCH	0.507	0.597	0.639	0.683	0.703

Note. The table shows the loss values of the root mean absolute error function for the prediction of the realized variance for all forecast horizons. The out-of-sample forecasts cover 2010M1-2018M1.

Lastly, the RMSD and the RMAD are considered and can be found in Appendix C. Both give similar results, where the LLF again gives superior results compared to the competitor and GARCH- MIDAS models. The HAR with leverage is the only model that comes close to the LLF model with all variables. However, their values are outperformed by the LLF models.

Next to comparative models of the GARCH-MIDAS and the LLF, the SHAP values were introduced in Section 4.5. First, the variable importance is plotted, which is easily interpretable. Then, the average SHAP values are presented for each feature from high to low.

Figure 1

The variable importance based on SHAP of all features for the prediction on 2010-01-04
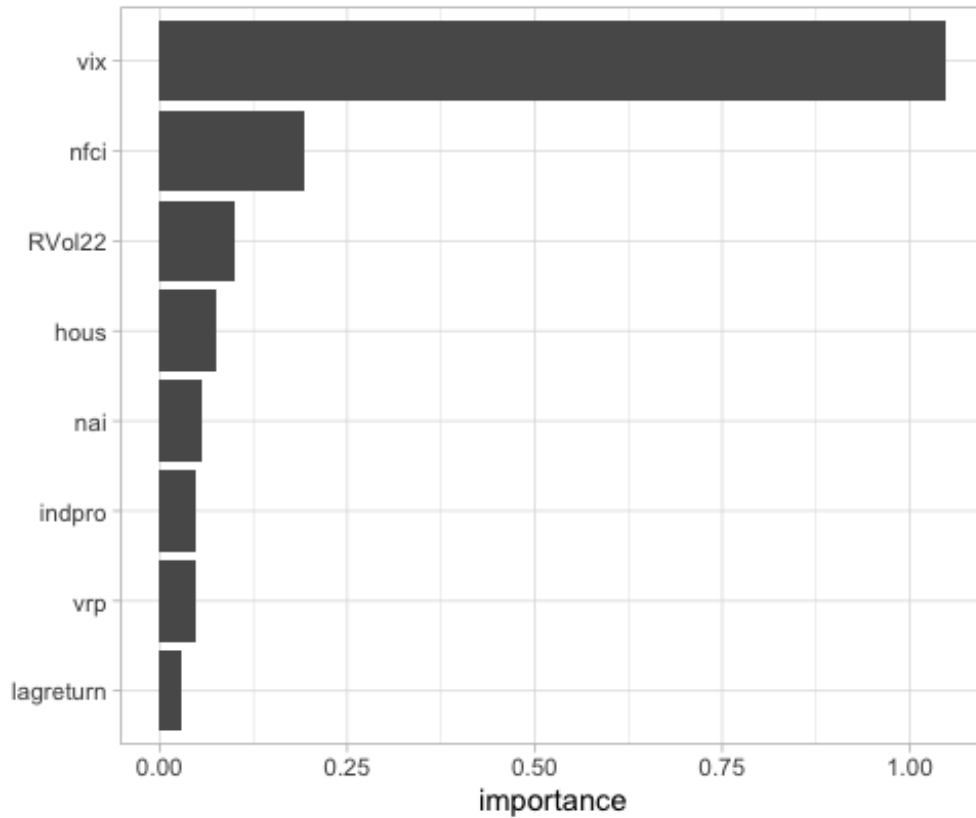
The plot above suggests that the VIX is the most important feature in the model. It contributes by far the most, after which is the NFCI with much lower importance than VIX. In third place is the average realized volatility of the previous 22 days (RVol22). In last comes the lag return, which confirms the results from the loss functions. The average losses stayed the same with and without the inclusion of this variable in the model.

One disadvantage of the variable importance is that we cannot see if a variable contributes positive or negative values to the target variable. Hence, the SHAP is plotted next to show overall metrics.

Figure 2

The SHAP variable importance plot of all features for the prediction on 2010-01-04
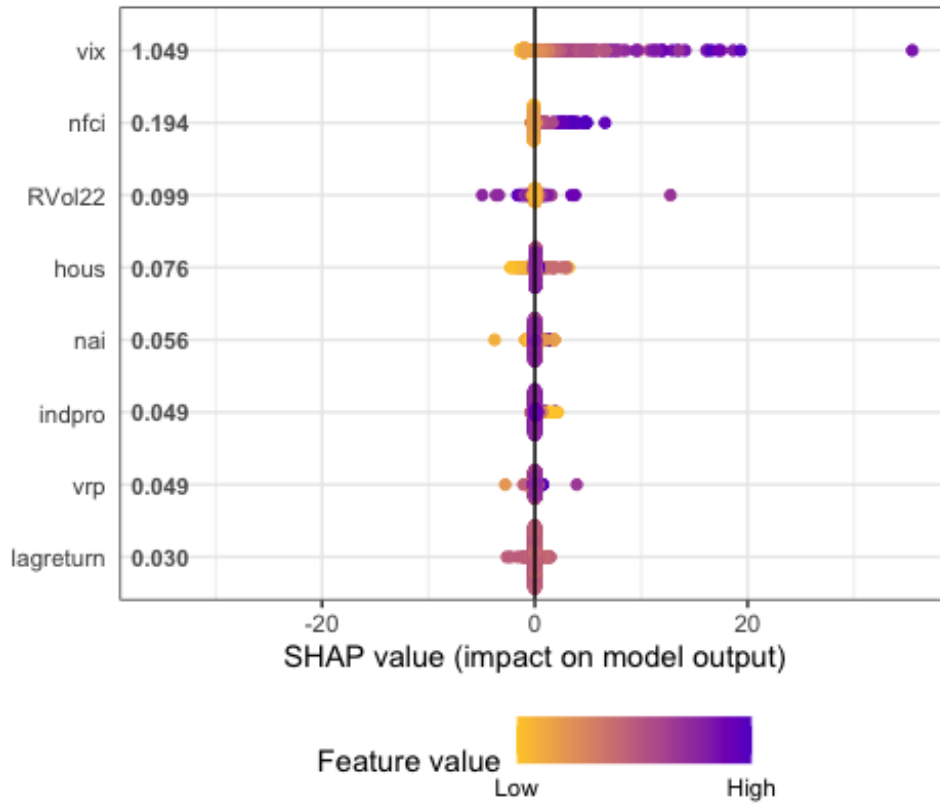
Here, the plot shows that a high value of the VIX has a high positive effect on the realized variance. Contrary, a high value of the RVol22 can have a high negative and positive effect. There are no features with only a high negative impact. Clearly, high values of the variables from housing starts to the lag return do not significantly affect either direction.

To summarize, the LLF method using all macroeconomic variables mentioned in Section 3 shows the lowest losses, where the use of LASSO does not seem to decrease the QLIKE losses. However, the mean squared prediction errors do decrease slightly with the use of LASSO for prediction. The difference in QLIKE losses between LLF methods and the GARCH- MIDAS increases up to almost ten times for the longer forecast horizons. For all other losses, the LLF models including all variables outperform all GARCH-MIDAS and competitor models. In addition, the macroeconomic variables which have the most effect on the predictions are VIX, NFCI, and RVol22.

6 Conclusion

This paper introduces and debates the features of a series of multiplicative volatility models, including the GARCH-MIDAS and local linear forests. This is done through an empirical application on the realized variance of S&P 500 stock returns retrieved from Conrad and Kleen (2019). Their results showed that the GARCH-MIDAS model performed best for forecast horizons 2- and 3-months-ahead. These results are then compared to the outcomes after applying the LLF method of Friedberg et al. (2020). Both QLIKE losses and mean squared prediction errors are considered as reference material.

The choice of covariates affects the performance of LLF models: only including (the lags of) housing starts in the direct prediction of realized variance did not improve the current models. Surprisingly, the LLF method including all macroeconomic variables performed best at all forecast horizons, such that the relative forecast performance does not depend on the length of the horizon. For these sets of models, the inclusion of the lag return as a variable showed to be relatively insensitive to the forecast outcomes; the same applies to the use of LASSO for variable selection in the prediction step. The best LLF specifications produce volatility estimates that are better than the GARCH-MIDAS forecasts. These findings imply that LLF models based on all macroeconomic data, including variables with different frequencies, may be more effective than the GARCH-MIDAS specification in forecasting financial volatility. Such improved forecasts could be used to better value options, and when used correctly, they can help optimize portfolio strategies.

However, it must be noted that the forecasts from the LLF models use the information on future macroeconomic variables which have not yet been attained. This makes the model less robust when applied in practice. Hence, further research is still to be done on forecasting the realized variances without knowing future macroeconomic variables. Due to the great results the LLF model has on predicting, it is useful to do more research on this model.

Lastly, the SHAP values were calculated to see which feature contributes the most to the target variable. The VIX index showed to be the most important, where high values of the VIX have a high positive effect. This result is to be expected as this index is a measure to estimate the expected volatility of the S&P 500. The objective is to estimate the realized variance; hence this feature is naturally important. The second variable is the National Financial Condition Index, where positive values indicate tighter financial conditions than average. High values of the NFCI correspond to a positive impact on the realized volatility. An explanation of this could be the decreasing money

supply. When the interest rates rise due to monetary policies of the central bank, it is more difficult to receive credit. Contrary, previous research did not directly relate stock market volatility and money supply (Taamouti, 2014; Choi and Yoon, 2015). However, the indirect effect of the money supply through macroeconomic features should not be overlooked. The third variable, which showed only slight importance to the target variable, is the backward measure of daily volatility. A large positive value of this variable can negatively or positively affect the realized volatility. This result also deviates from what is expected, as the negative effect is surprising. Further research upon the macroeconomic explanation of these outcomes would be useful for the economic interpretation of the obtained results.

References

Amit, Y., & Geman, D. (1997). Shape quantization and recognition with randomized trees. *Neural computation*, *9*(7), 1545–1588.

Asgharian, H., Hou, A. J., & Javed, F. (2013). The importance of the macroeconomic variables in forecasting stock return variance: A garch-midas approach. *Journal of Forecasting*, *32*(7), 600–612.

Athey, S., Tibshirani, J., Wager, S., et al. (2019). Generalized random forests. *Annals of Statistics*, *47*(2), 1148–1178.

Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, *31*(3), 307–327. https://doi.org/https://doi.org/10.1016/0304-4076(86)90063-1

Brandt, M. W., & Kang, Q. (2002). *On the relationship between the conditional mean and volatility of stock returns: A latent var approach* (Working Paper No. 9056). National Bureau of Economic Research. https://doi.org/10.3386/w9056

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees*. Taylor Francis.

Breiman, L. (1996). Bagging predictors. *Machine learning*, *24*(2), 123–140.

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5–32.

Choi, K.-H., & Yoon, S.-M. (2015). The effect of money supply on the volatility of korean stock market. *Modern Economy*, *06*, 535–543. https://doi.org/10.4236/me.2015.65052

Conrad, C., Custovic, A., & Ghysels, E. (2018). Long-and short-term cryptocurrency volatility components: A garch-midas analysis. *Journal of Risk and Financial Management*, *11*(2), 23.

Conrad, C., & Kleen, O. (2019). Two are better than one: Volatility forecasting using multiplicative component garch-midas models. *Journal of Applied Econometrics*, *35*. https://doi.org/10.1002/jae.2742

Conrad, C., & Loch, K. (2015). Anticipating long-term stock market volatility. *Journal of Applied Econometrics*, *30*(7), 1090–1114. https://doi.org/https://doi.org/10.1002/jae.2404

Corsi, F. (2009). A simple approximate long-memory model of realized volatility. *Journal of Financial Econometrics*, *7*(2), 174–196.

Corsi, F., & Renò, R. (2012). Discrete-time volatility forecasting with persistent leverage effect and the link with continuous-time volatility modeling. *Journal of Business Economic Statistics*, *30*(3), 368–380. http://www.jstor.org/stable/23243735

Cutler, D. R., Edwards Jr, T. C., Beard, K. H., Cutler, A., Hess, K. T., Gibson, J., & Lawler, J. J. (2007). Random forests for classification in ecology. *Ecology*, *88*(11), 2783–2792.

Dorion, C. (2016). Option valuation with macro-finance variables. *Journal of Financial and Quantitative Analysis*, *51*(4), 1359–1389.

Engle, R., & Lee, G. (1999). A long-run and short-run component model of stock return volatility. *Engle, R.F. and White, H., Eds., Cointegration, Causality, and Forecasting: A Festschrift in Honor of Clive W.J. Granger*, 475–497.

Engle, R., Ghysels, E., & Sohn, B. (2013). Stock Market Volatility and Macroeconomic Fundamentals. *The Review of Economics and Statistics*, *95*(3), 776–797. https://doi.org/10.1162/REST_a_00300

Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica: Journal of the econometric society*, 987–1007.

Fang, L., Qian, Y., Chen, Y., & Yu, H. (2018). How does stock market volatility react to nvix? evidence from developed countries. *Physica A: Statistical Mechanics and its Applications*, *505*. https://doi.org/10.1016/j.physa.2018.03.039

Friedberg, R., Tibshirani, J., Athey, S., & Wager, S. (2020). Local linear forests. *Journal of Computational and Graphical Statistics*, 1–15. https://doi.org/10.1080/10618600.2020.1831930

Gama, J. (2004). Functional trees. *Machine Learning*, *55*(3), 219–250.

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*(1-2), 59–95.

Glosten, L. R., Jagannathan, R., & Runkle, D. E. (1993). On the relation between the expected value and the volatility of the nominal excess return on stocks. *The journal of finance*, *48*(5), 1779–1801.

Hamilton, J. D., & Lin, G. (1996). Stock market volatility and the business cycle. *Journal of applied econometrics*, *11*(5), 573–593.

Hansen, P. R., Lunde, A., & Nason, J. M. (2011). The model confidence set. *Econometrica*, *79*(2), 453–497. http://www.jstor.org/stable/41057463

Hansen, P. R., Huang, Z., & Shek, H. H. (2012). Realized garch: A joint model for returns and realized measures of volatility. *Journal of Applied Econometrics*, *27*(6), 877–906.

Hothorn, T., Lausen, B., Benner, A., & Radespiel-Tröger, M. (2004). Bagging survival trees. *Statistics in medicine*, *23*(1), 77–91.

Karalic, A. (1992). Linear regression in regression tree leaves. *In Proceedings of ECAI-92*, 440–441.

Kuhn, M., & Johnson, K. (2013). *Applied predictive modeling*. Springer.

Laurent, S., Rombouts, J. V., & Violante, F. (2013). On loss functions and ranking forecasting performances of multivariate volatility models. *Journal of Econometrics*, *173*(1), 1–10.

Liu, L., Patton, A., & Sheppard, K. (2015). Does anything beat 5-minute rv? a comparison of realized measures across multiple asset classes. *Journal of Econometrics*, *187*(1), 293–311. https://EconPapers.repec.org/RePEc:eee:econom:v:187:y:2015:i:1:p:293-311

Lundberg, S. (2019). Shap.

Lundberg, S., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *CoRR*, *abs/1705.07874*. http://arxiv.org/abs/1705.07874

Luong, C., & Dokuchaev, N. (2018). Forecasting of realised volatility with the random forests algorithm. *Journal of Risk and Financial Management*, *11*(4). https://doi.org/10.3390/jrfm11040061

Medeiros, M. C., Vasconcelos, G. F. R., Veiga, Á., & Zilberman, E. (2021). Forecasting inflation in a data-rich environment: The benefits of machine learning methods. *Journal of Business & Economic Statistics*, *39*(1), 98–119. https://doi.org/10.1080/07350015.2019.1637745

Meinshausen, N., & Ridgeway, G. (2006). Quantile regression forests. *Journal of Machine Learning Research*, *7*(6).

Menze, B. H., Kelm, B. M., Splitthoff, D. N., Koethe, U., & Hamprecht, F. A. (2011). On oblique random forests. *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 453–469.

Opitz, D., & Maclin, R. (1999). Popular ensemble methods: An empirical study. *Journal of artificial intelligence research*, *11*, 169–198.

Opschoor, A., van Dijk, D., & van der Wel, M. (2014). Predicting volatility and correlations with financial conditions indexes. *Journal of Empirical Finance*, *29*, 435–447.

Oshiro, T. M., Perez, P. S., & Baranauskas, J. A. (2012). How many trees in a random forest? In P. Perner (Ed.), *Machine learning and data mining in pattern recognition* (pp. 154–168). Springer Berlin Heidelberg.

Perez-Quiros, G., & Timmermann, A. (2000). Firm size and cyclical variations in stock returns. *The Journal of Finance*, *55*(3), 1229–1262.

Polamuri, S., Srinivas, K., & Mohan, A. (2019). Stock market prices prediction using random forest and extra tree regression. *International Journal of Recent Technology and Engineering*, *8*, 1224–1228. https://doi.org/10.35940/ijrte.C4314.098319

Probst, P., & Boulesteix, A.-L. (2017). To tune or not to tune the number of trees in random forest? *Journal of Machine Learning Research*, *18*.

Rusch, T., & Zeileis, A. (2013). Gaining insight with recursive partitioning of generalized linear models. *Journal of Statistical Computation and Simulation*, *83*(7), 1301–1315.

Sadorsky, P. (2021). A random forests approach to predicting clean energy stock prices. *Journal of Risk and Financial Management*, *14*(2). https://doi.org/10.3390/jrfm14020048

Schwert, G. W. (1989). Why does stock market volatility change over time? *The journal of finance*, *44*(5), 1115–1153.

Shephard, N., & Sheppard, K. (2010). Realising the future: Forecasting with high-frequency-based volatility (heavy) models. *Journal of Applied Econometrics*, *25*(2), 197–231.

Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., & Feuston, B. P. (2003). Random forest: A classification and regression tool for compound classification and qsar modeling. *Journal of chemical information and computer sciences*, *43*(6), 1947–1958.

Taamouti, A. (2014). Stock market's reaction to money supply: A nonparametric analysis. *Studies in Nonlinear Dynamics and Econometrics*, *19*. https://doi.org/10.1515/snde-2013-0059

Tibshirani, J., Athey, S., Friedberg, R., Hadad, V., Miner, L., Wager, S., & Wright, M. (2019). *Grf: Generalized random forests (beta)* [R Package Version 0.10.3].

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, *58*(1), 267–288.

Torgo, L. (1997). Functional models for regression tree leaves. *ICML*, *97*, 385–393.

Tyralis, H., & Papacharalampous, G. (2017). Variable selection in time series forecasting using random forests. *Algorithms*, *10*(4), 114.

Vassallo, D., Krishnamurthy, R., Sherman, T., & Fernando, H. J. (2020). Analysis of random forest modeling strategies for multi-step wind speed forecasting. *Energies*, *13*(20), 5488.

Wager, S., & Athey, S. (2018). Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association*, *113*(523), 1228–1242.

Whitelaw, R. (1994). Time variations and covariations in the expectation and volatility of stock market returns. *The Journal of Finance*, *49*(2), 515–541. https://doi.org/https://doi.org/10.1111/j.1540-6261.1994.tb05150.x

Wu, Z., & Chong, T. (2021). Does the macroeconomy matter to market volatility? evidence from us industries. *Empirical Economics.* https://doi.org/10.1007/s00181-020-02001-3

Xu, R., Nettleton, D., & Nordman, D. J. (2016). Case-specific random forests. *Journal of Computational and Graphical Statistics, 25*(1), 49–65.

Zeileis, A., Hothorn, T., & Hornik, K. (2008). Model-based recursive partitioning. *Journal of Computational and Graphical Statistics, 17*(2), 492–514.

Appendices

A   Summary Statistics

Table 6

Summary statistics of the data set

	Freq.	Start	Min.	Max.	Mean	Median	SD	Skew	Kurt.	Obs.
Stock market data										
S&P 500 return	d	1971	-22.900	10.957	0.028	0.044	1.062	-1.039	28.811	11938
$\sqrt{(RV)}$	d	2000	0.127	8.842	0.872	0.716	0.604	3.224	21.930	4600
RVol(22)	d	1989	0.234	5.542	0.949	0.801	0.560	2.973	17.465	7390
Explanatory Variables										
VIX	d	1990	0.576	5.094	1.219	1.102	0.495	2.085	10.628	7135
NFCI	w	1973	-0.990	4.670	0.002	-0.330	1.001	1.944	6.533	2470
NAI	m	1971	-5.160	2.760	-0.004	0.060	1.003	-1.212	6.961	568
Δ IP	m	1971	-4.430	2.375	0.181	0.219	0.724	-1.220	8.818	568
Δ Hous	m	1971	-30.674	25.672	-0.068	-0.191	8.029	-0.034	3.775	568

B GARCH-MIDAS results

Table 7

GARCH-MIDAS estimation results: one variable

	α	β	γ	m	θ	w_1	w_2	K	LLH	BIC	VR(X)
Daily τ_t											
VIX	0.000	0.853***	0.095***	-2.129***	1.524***	1	3.470**	3	-9.138	18.339	76.14
	(0.010)	(0.021)	(0.015)	(0.086)	(0.067)		(1.371)				
Monthly τ_t											
Δ Hous	0.019***	0.897***	0.119***	-0.079	-0.237***	1.695***	2.586***	36	-14.449	29.192	19.63
	(0.005)	(0.005)	(0.007)	(0.076)	(0.034)	(0.383)	(0.770)				

Note. Estimation results of GARCH-MIDAS for single explanatory variables VIX and Δ Housing are reported above, starting from 1990M1 and 1974M1 respectively. Bollerslev-Woolridge standard errors are given in parentheses below, with significance levels *10%, **5% and ***1%. Only for VIX, a restricted weighting scheme is chosen; $w_1 = 1$. LL denotes the maximized log-likelihood function, BIC stands for Bayesian Information Criterion and VR(X) is the variance ratio ($= \text{var}(\log(\tau_M^X))/\text{var}(\log(\sigma_M^X))$). Estimates of μ are omitted.

Table 8

GARCH-MIDAS estimation results: two variables, VIX plus other

	α	β	γ	m	θ^X	w_1^X	w_2^X	θ^{VIX}	w_2^{VIX}	K^X	LLH	BIC	VR(VIX, X)
Daily τ_t													
VIX	0.000	0.853***	0.095***	-2.129***	-	-	-	1.524***	3.470**	3	-9.138	18.339	76.14
	(0.010)	(0.021)	(0.015)	(0.086)				(0.067)	(1.371)				
	0.000	0.852***	0.099***	-1.993***	0.118	1	2.252	1.451***	3.617**	52	-9.110	18.300	75.84
	(0.010)	(0.020)	(0.016)	(0.143)	(0.085)		(4.152)	(0.093)	(1.518)				
Monthly τ_t													
NAI	0.000	0.870***	0.092***	-2.032***	-0.108**	1	119.372	1.431***	3.775**	36	-9.133	18.346	75.06
	(0.009)	(0.018)	(0.015)	(0.100)	(0.046)		(326.330)	(0.079)	(1.594)				
Δ IP	0.000	0.876***	0.084***	-2.133***	-0.043	1	8.960	1.528***	3.806**	36	-9.139	18.357	75.91
	(0.009)	(0.018)	(0.014)	(0.096)	(0.089)		(34.803)	(0.072)	(1.520)				
Δ Hous	0.000	0.863***	0.097***	-2.035***	-0.061**	1.001	2.139	1.446***	3.605**	36	-9.135	18.359	74.99
	(0.009)	(0.019)	(0.015)	(0.094)	(0.024)	(0.743)	(2.462)	(0.074)	(1.503)				

Note. Estimation results of GARCH-MIDAS for combination of daily VIX and NAI, Δ IP, and Δ Housing as explanatory variables, starting from 1990M1 to 2018M4 respectively. For comparison, the estimation results of the single explanatory variable VIX are included in the table. All parameters with superscript X are related to the second explanatory variable. Bollerslev-Woolridge standard errors are given in parentheses below, with significance levels *10%, **5% and ***1%. Only for VIX, a restricted weighting scheme is chosen; $w_1 = 1$. LL denotes the maximized log-likelihood function, BIC stands for Bayesian Information Criterion and VR(X) is the variance ratio ($= \text{var}(\log(\tau_M^X))/\text{var}(\log(\sigma_M^X))$). Estimates of μ are omitted.

C   RMSD and RMAD results

Table 9

Out-of-sample root mean standard deviation values for all forecast horizons

RMSD		**1d**	**2w**	**1m**	**2m**	**3m**
LLF	Δ Hous	0.490	0.560	0.554	0.533	0.593
	All var	0.186	0.188	0.189	0.188	0.188
	All var - LASSO	0.184	0.186	**0.187**	0.187	0.188
	All var + $r_{i,t-1}$	0.186	0.187	0.190	0.187	0.188
	All var + $r_{i,t-1}$ - LASSO	**0.184**	**0.186**	0.188	**0.186**	**0.187**
GARCH-MIDAS	VIX	0.244	0.311	0.331	0.354	0.366
	Δ Hous	0.265	0.333	0.365	0.391	0.396
	VIX and NFCI	0.240	0.306	0.325	0.346	0.357
	VIX and NAI	0.243	0.309	0.329	0.352	0.364
	VIX and Δ IP	0.242	0.309	0.329	0.352	0.364
	VIX and Δ Hous	0.247	0.314	0.335	0.358	0.370
Competitors	HAR	0.205	0.297	0.342	0.399	0.433
	HAR (leverage)	0.196	0.297	0.343	0.399	0.434
	Real GARCH	0.207	0.280	0.312	0.348	0.363

Note. The table shows the loss values of the root mean standard deviation function for the prediction of the realized variance for all forecast horizons. The out-of-sample forecasts cover 2010M1-2018M1.

Table 10

Out-of-sample root mean absolute deviation values for all forecast horizons

RMAD		**1d**	**2w**	**1m**	**2m**	**3m**
LLF	Δ Hous	0.649	0.700	0.691	0.681	0.706
	All var	0.386	0.390	0.389	0.390	**0.389**
	All var - LASSO	0.385	0.389	**0.388**	0.389	0.390
	All var + $r_{i,t-1}$	0.387	0.389	0.390	0.389	0.390
	All var + $r_{i,t-1}$ - LASSO	**0.384**	**0.389**	0.390	**0.388**	0.390
GARCH-MIDAS	VIX	0.452	0.508	0.527	0.545	0.554
	Δ Hous	0.472	0.533	0.563	0.585	0.588
	VIX and NFCI	0.450	0.505	0.525	0.541	0.549
	VIX and NAI	0.452	0.507	0.526	0.544	0.613
	VIX and Δ IP	0.451	0.507	0.526	0.544	0.553
	VIX and Δ Hous	0.455	0.511	0.532	0.550	0.559
Competitors	HAR	0.403	0.495	0.538	0.586	0.613
	HAR (leverage)	0.395	0.494	0.539	0.585	0.614
	Real GARCH	0.405	0.479	0.515	0.548	0.561

Note. The table shows the loss values of the root mean absolute deviation function for the prediction of the realized variance for all forecast horizons. The out-of-sample forecasts cover 2010M1-2018M1.

D   Code Description

This paper uses different packages to obtain the results, divided into a total of nine scripts. Hence, a short explanation of these scripts' order is given here so that the results can be reproduced.

First, the MasterThesis.R needs to be run, containing all the necessary packages for the whole project. It also creates empty files where the models and results are stored. After which, the models can be estimated using the estimate_models_real_time.R script. It first reads the data downloaded from the Journal of Applied Econometrics data archive, following the paper of Conrad and Kleen (2019). With this data, the models are estimated individually, starting with the LLF models, then the GARCH-MIDAS, and ending with the competitor models. Note that most of this code has been copied by Conrad and Kleen (2019) in order to reproduce their results.

Before the forecasts can be made, the LASSO.R script is run to get the selected variables. The results of these estimations are shown in Table 1. After obtaining the models and LASSO selected variables, forecasts are made with the make_forecasts.R script. First, the parameters of the GARCH-MIDAS and competitor models need to be retrieved before forecasting. Then a data frame is created where all forecasts are stored for later use. After this, one big forecast loop begins. Again, most of this builds upon the existing code of Conrad and Kleen (2019).

Lastly, there are different scripts for different performance measures. The qlike.R together with the MCS.R script produce Tables 2 and 3. The qlike.R script begins with some alterations to the LLF forecasts, as the realized variance cannot be equal to one. These values are replaced with a very low number (this cannot be equal to zero; otherwise, this will produce a NaN for qlike). The rest of the code builds upon the available scripts of Conrad and Kleen (2019). The mse.R script contains the estimations of all four mean prediction errors, to make Tables 4, 5, 9, and 10. Ending with the SHAP values mentioned in Section 4.5. The shap_analysis.R uses the shap.R script, which contains different functions made by Lundberg (2019). His shap_analysis.R script has been modified to the data in this paper. With these scripts Figures 1 and 2 are made. For a more detailed description, see the scripts.