ERASMUS UNIVERSITY
ERASMUS SCHOOL OF ECONOMICS

BACHELOR THESIS
ECONOMETRICS AND OPERATIONS RESEARCH

The Joint Lasso for realized volatility modelling using HAR

Author: Platenkamp, M. *Student ID number:* 482261
Supervisor: Tetereva, A. *Second assessor:* Wang, W.

July 11, 2020

Abstract

This thesis assesses the performance of the Joint Lasso by Dondelinger, Mukherjee, and Initiative (2020). This framework is used for modelling the the diagonal elements of the covariance matrix of 10 different assets listed on the NYSE with the Heterogeneous Autoregressive model of Realized Volatility (HAR). The predictive power of the constructed models are reported for several sample periods from the data set from Noureldin, Shephard, and Sheppard (2012). The results suggest that the performance of the multivariate HAR model can be improved by using the Joint Lasso instead of equation-by-equation OLS.

Contents

1 Introduction

It is widely known that the daily returns of financial assets, especially of stocks are difficult if not impossible to predict, although the volatility of the returns seems to be relatively easier to forecast (Barndorff-Nielsen and Shephard 2002). Modelling volatility could be of interest to people who are interested in financial sectors. In particular in risk management, portfolio management and asset pricing, volatility modelling can be of great importance (Čech and Baruník 2017). Studying the interrelations of volatility, measured over different time horizons, allows one to reveal the dynamics of the different market components (Corsi 2009). For example, it is shown by Sarwar (2003) that there is a presence of strong contemporaneous positive feedbacks between the exchange rate volatility and the trading volume of call and put options.

As is stated by Corsi (2009), financial data present a series of well-known stylized facts that pose serious challenges to standard econometric models, so it would be appropriate to search for the best possible model to capture this volatility. To capture the slowly decaying autocorrelation feature of realized volatility also known, and referred to by Čech and Baruník (2017), as long-memory, several models are proposed. Corsi (2009) proposes the Heterogeneous Autoregressive model of Realized Volatility (HAR). Audrino and Knaus (2016) state that this model enjoys great popularity, which can be explained by the simplicity of the model its structure. Corsi (2009) claims with the HAR model that tomorrow's realized volatility is a sum of daily, weekly, and monthly averages of realized volatilities. Corsi, Audrino, and Renó (2012) describe numerous variants of the standard HAR model, but in this thesis the focus is only on the standard model, since in general differences in predictive power are not found to be statistically significant compared to variants of this model (Audrino, Huang, and Okhrin 2018). As stated in Vortelinos (2017), this simple model is not outperformed by non-linear models, by saying that Principal Components Combining, neural networks and GARCH are not more accurate than the Heterogeneous Autoregressive (HAR) model.

This simple model is suitable for univariate modelling, as it only models the realized volatility of one asset. The extension on multivariate modelling is used by Chiriac and Voev (2011). They model realized covariance matrices by modelling a vector of Cholesky factors, which is later explained in the methodology section of this thesis. A disadvantage of this approach is described by Čech and Baruník (2017). They state that a significant amount of information is left in the error term and that, more importantly, common structures of the realized volatility may be left unmodelled. To deal with this problem they propose a system of seemingly unrelated HAR regressions. The idea is that they construct a multivariate HAR model which will capture possible common structure. The model proposed by Čech and Baruník (2017) is named GHAR and provides us with a more efficient estimator that accounts for otherwise hidden dependencies among variables. Furthermore, they show that the GHAR-model is not systematically dominated by any benchmark model.

Concluding from the performance of the GHAR-model it could be beneficial to model dependencies between realized volatilities of different assets. A way to use these dependencies in a multivariate setting is estimating the parameters of the HAR-model or GHAR-model with the "Joint Lasso". Dondelinger, Mukherjee, and Initiative (2020) consider high-dimensional regression over subgroups of observations. They do this in a penalized framework and named it the "Joint Lasso". They show that this method can improve performance relative to pooling or subgroup-wise analysis. In this thesis the realized volatilities of various subgroups of stocks will be modelled by using the Joint Lasso to estimate the parameters of the multivariate HAR model. Like Dondelinger, Mukherjee, and Initiative (2020), this thesis considers the case where models are constructed for sub-groups of assets, and each subgroup-specific regression model is expected to be similar but not necessarily identical. So for each subgroup of assets, regression coefficients are estimated so that each asset in a subgroup is given the same regression coefficients. Summarizing, the main question this thesis tries to answer: "Can the predictive performance of the multivariate HAR model for diagonal elements of the covariance matrix be improved with the Joint Lasso?" A sub-question this thesis tries to answer: "Can we reduce the rolling window size using the Joint Lasso to improve the performance in economic turbulent times?"

Ray and Tsay (2000) investigate the strength of long-range dependencies in stock volatilities for different groups of S&P500 companies. They show that almost all of the companies analyzed exhibit strong persistence in volatility. Furthermore, they study the effects of company size and sector. Moreover, their results indicate the existence of some size effects and that the effects of company sector are pronounced. Kumar (2017) implements a so called "HAR-DL" model for various equity sectors (Automobiles, Financials, Industrial, Telecom and Pharmaceutical). This model is an expansion of the standard HAR model and it is shown that a different HAR-DL model is appropriate for each equity sector. This thesis will investigate the realized volatilities of subgroups of companies in a similar spirit.

The structure of the thesis is as follows. First, the literature on the topics of realized volatility and estimation methods such as OLS and Lasso is discussed. Next to this, the data set that is used is described in detail and it is further described how the data is used. Furthermore, the results that are collected through the data are displayed in the results section. Finally, some interesting conclusions are drawn from the results in the conclusion section.

2 Literature review

Robert Tibshirani (1996) uses the good features of both the ridge regression and subset selection to obtain a more accurate model for making predictions. This method, so called Lasso, is an alternative for the Ordinary Least Squares (OLS) regression and uses a bias-variance trade-off to improve the prediction accuracy of the OLS estimates, since OLS normally has low bias, but large variance (Robert Tibshirani 1996).

The Lasso, as proposed by Robert Tibshirani (1996) is used in numerous variants and applications. For example, Audrino and Knaus (2016) use Lasso and its statistical inference theory to see whether the lag structure (1, 5, 22) implied from an economic point of view for the HAR model can be recovered by statistical methods. They fail however, to show strong evidence that such a fixed lag structure can be recovered by a flexible model.
Audrino and Camponovo (2013) use the adaptive lasso (Zou 2006) to perform inference in classical time series model. They state that the method proposed produces efficient parameter estimations for time series models, which is appropriate for the the HAR model, as it also is a time series model.

Another variant of the Lasso regression is estimating parameters for subgroups of observations. Dondelinger, Mukherjee, and Initiative (2020) propose the "Joint Lasso" to jointly estimate the regression coefficients of subgroups that "induces global sparsity and encourages similarity between subgroup-specific coefficients". They state that the advantage of this method is that it accounts for different underlying regression models for each specified subgroup, but in contrast to the subgroup-wise approach it takes advantage of similarities between subgroups. The hypothesis is that this method improves the model accuracy when considering subgroups of observations.
Dondelinger, Mukherjee, and Initiative (2020) make estimations and predictions of cognitive scores for Alzheimer's disease, disease progression of ALS and therapeutic response in cancer cell lines. This can be done with the same models that are used for predicting financial data, as an dependent variable (such as disease progression) is estimated with independent variables (such as patient information). Different types of diseases are put together in a different groups, which also can be done with different financial data.
Thus, for all these examples the data sample can be divided in subgroups, and for the underlying regression model of each subgroup the same explanatory variables (p-features) are used. To conclude, all the examples of the Joint Lasso stated above demonstrate the gains joint estimation can offer in prediction as well as in providing subgroup-specific information.

3 Methodology

3.1 Realized volatilty

As stated in the previous sections of this thesis, it is desirable to model the realized volatility at time t. One can use intraday transactions data to calculate the realized volatility of an asset. The standard definition of the realized volatility over a time interval of one day is denoted by Corsi (2009) as:

$$RV_t^{(d)} = \sqrt{\sum_{j=0}^{M-1} r_{t-j*\Delta}^2}, \tag{1}$$

Where $\Delta = \frac{1d}{M}$, and $r_{t-j*\Delta}$ denotes the continuously compounded Δ-frequency returns, that is, the intraday return sampled at interval Δ (the subscript t indexes the day, j indexes the time within day t).

Next to the basic measure for $RV_t^{(d)}$, the Average estimator $RV_t^{(d)}(AVG)$, the Two-Scale estimator $RV_t^{(d)}(TS)$ (Zhang, Mykland, and Aıt-Sahalia 2005), and the Realized Kernel $RV_t^{(d)}(ker)$ (Barndorff-Nielsen, Peter Reinhard Hansen, et al. 2008) can be applied. The measurement used for realized volatility has large implications for its magnitude as well as its predictability. For example, the Realized Kernel from Barndorff-Nielsen, Peter Reinhard Hansen, et al. (2008) is defined as:

$$RV_t^{(ker)} = \sum_{h=-n}^{n} K(\frac{h}{H})\Gamma_h, \tag{2}$$

where $\Gamma_h = \sum_{j=h+1}^{n} r_{t,j} r_{t,j-|h|}$, and where K(x) is a Kernel weight function K(0)=1 and K'(0) = 0. The use of the Realized Kernel as a measure for the realized volatility can be defended by the claim that it is noise-robust (Noureldin, Shephard, and Sheppard 2012).

3.2 Microstructure noise

Realized volatility is a latent variable and in the literature high-frequency data is used to calculate this variable. High-frequency data brings the problem of microstructure noise and gives a substantial bias to the daily volatility measurement (Andersen, Bollerslev, and Meddahi 2011). High-frequency financial data consists mostly of asset trade prices, which lie between the bid and ask prices. In a market without volatility a price for an asset could be equal to \$50 during the whole day. However, market costs are present and influence the price, so the actual ask and bid price could be \$50.10 and \$49.90 respectively for instance. The observed price in the data collected will be between the bid and ask price. This effect will bias the modelling of the volatility (Andersen, Bollerslev, and Meddahi 2011).
Using lower frequency data reduces the effect of microstructure noise, although the modelling of the realized volatilities is less accurate (Noureldin, Shephard, and Sheppard 2012). Further details about cleaning the noise from the data can be found in the data section.

3.3 HAR model

Corsi (2009) defines the HAR model for realized volatility as a hierarchical model with only three components corresponding to the realized volatility over different time horizons. The proposed model is written as follows:

$$RV_{t+1d}^{(d)} = c + \beta^{(d)} RV_t^{(d)} + \beta^{(w)} RV_t^{(w)} + \beta^{(m)} RV_t^{(m)} + \omega_{t+1d}, \tag{3}$$

where $RV_t^{(d)}$ is the realized volatility over a time interval of one day at time t. ω_{t+1d} is the contemporaneously and serially independent zero-mean innovation. The weekly realized volatility at time t is defined as:

$$RV_t^{(w)} = \frac{1}{5}(RV_t^{(d)} + RV_{t-1d}^{(d)} + ... + RV_{t-4d}^{(d)}). \tag{4}$$

And the monthly realized volatility is defined as:

$$RV_t^{(m)} = \frac{1}{22}(RV_t^{(d)} + RV_{t-1d}^{(d)} + ... + RV_{t-21d}^{(d)}) \tag{5}$$

The HAR-model stated above is suitable for the prediction of the realized volatility of one asset. Predicting the realized volatility in a multivariate setting for several assets, the covariance matrix is needed. This covariance matrix is really important for portfolio management and asset pricing (Chiriac and Voev 2011). Therefore, Chiriac and Voev (2011) propose a method to estimate multivariate volatility models. Firstly, Cholesky decomposition is used for the facotrization of the covariance matrix. In the next step the Cholesky series can be forecasted using the HAR model as a suitable time series model. In the last step the covariance matrix can be reconstructed.

The operation "vech" is used to factorize the covariance matrix as follows:

$$X_t = vech(P_t), \tag{6}$$

where P_t are the Cholesky factors, so that $P_t P_t' = \Sigma_t$ and X_t is the $m \times 1$ vector with m $= \frac{q(q+1)}{2}$, denoting the indexes of the elements of the lower triangular covariance matrix. The covariance matrix is obtained by the inverse vech operation. In this thesis, only the realized variances (diagonal elements of the covariance matrix) are used, and this is further explained in the data section. However, Chiriac and Voev (2011) use all elements of the covariance matrix in their model, as it becomes:

$$X_{t+1}^{(d)} = c + \beta^{(d)} X_t^{(d)} + \beta^{(w)} X_t^{(w)} + \beta^{(m)} X_t^{(m)} + \omega_{t+1d} \quad with \quad \omega \ i.i.d. \tag{7}$$

This model can be constructed for each X at time t, which can be rewritten to equation (3) if only the diagonal elements are used. Čech and Baruník (2017) came with an extension and built a system of seemingly unrelated HAR regressions for all elements of X_t. They state that the advantage of this approach is that estimating a multivariate HAR model will capture the separate dynamics of the variances and covariances, and also possible common structure. As known from Corsi (2009) error terms from HAR are heteroscedastic with less efficient coefficient estimates. As there is no information about dependence between equations left in the method proposed by Chiriac and Voev (2011), Čech and Baruník (2017) estimate the following system:

$$\begin{pmatrix} X_{1,t+1}^{(d)} \\ \vdots \\ X_{m,t+1}^{(d)} \end{pmatrix} = \begin{bmatrix} x_{1,t} & & 0 \\ & \ddots & \\ 0 & & x_{m,t} \end{bmatrix} \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_m \end{pmatrix} + \begin{pmatrix} \omega_{1,t} \\ \vdots \\ \omega_{m,t} \end{pmatrix} \tag{8}$$

The model is rewritten in the form:

$$X = Z\beta + \omega \tag{9}$$

With Z being a block diagonal matrix containing all the X. Čech and Baruník (2017) estimate the model parameters with two-step feasible generalized least squares. Another way of estimating the parameters is by equation-by-equation OLS. This estimation method is also neglecting dependencies between the equation, but in this thesis it will be considered as the "benchmark". Note again that all the X in equation (8) can be replaced by RV from equation (3) if only diagonal elements are used. In the next section another estimation method is discussed for the coefficients that also uses common information of several X_t. The hypothesis is that this estimation method will also outperform equation-by-equation OLS.

3.4 The Joint Lasso

The constant and the coefficients in equation (3) can be estimated by using Ordinary Least Squares (OLS). Another already discussed regression method for estimating these parameter is the Lasso regression. Using the classic Lasso regression (Robert Tibshirani 1996) for estimating the realized volatility looks as follows:

$$\widehat{B} = \underset{B=[\beta_1 \dots \beta_k]}{argmin} \left\{ ||RV_{t+1}^{(d)} - c - RV_t^{(d+w+m)}\beta||_2^2 + \lambda||\beta||_1 \right\} \tag{10}$$

In the equation above the two terms displayed are minimized. The first term denotes the sum of squared residuals that should be minimized and the second term denotes a penalty. The tuning parameter λ denotes how "severe" the penalty is, and the penalty itself tends to include or exclude all betas or shrink them (Robert Tibshirani 1996). Lasso can be used for several other purposes, and in the setup of this thesis it is used in the penalized framework from Dondelinger, Mukherjee, and Initiative (2020). This is an extension of the classic lasso function and is denoted as:

$$\widehat{B} = \underset{B=[\beta_1 \dots \beta_k]}{argmin} \sum_{k=1}^{k} \left\{ \frac{1}{n_k} ||RV_{k,t+1}^{(d)} - c - RV_{k,t}^{(d+w+m)}\beta_k||_2^2 + \lambda||\beta_k||_1 + \gamma \sum_{k'>k} \tau_{k,k'}||\beta_k - \beta_{k'}||_2^2 \right\} \tag{11}$$

If the formulation above is compared with the classic lasso in equation (10) several differences are noted and are also described by Dondelinger, Mukherjee, and Initiative (2020). First of all, they describe that the betas are separated in **k** subgroups, so for each subgroup different coefficients are obtained. Next to this, they mention that another penalty (the fusion penalty) is added to the formulation. They also state that for each subgroup **k**, the difference is taken between entire vectors of subgroup-specific coefficients to penalize differences between. They explain that this penalty encourages similarity between the subgroups, so information of similar subgroups can be shared.

In the formulation above from Dondelinger, Mukherjee, and Initiative (2020), the number of subgroups is denoted as **k**, so for each subgroup **k** a vector of the coefficients $\beta^{(d)}$, $\beta^{(w)}$ and $\beta^{(m)}$ exist which are all elements of the matrix β_k with dimension $[3 \times n_k]$. The vector $RV_{k,t+1}^{(d)}$ denotes the dependent variable of the regression, containing all the realized volatilities of subgroup **k**. The matrix $RV_{k,t}^{(d+w+m)}$ denotes the explanatory variable containing all the AR-features $RV_t^{(d)}$, $RV_t^{(w)}$, $RV_t^{(m)}$ and a constant. The dimensions for $RV_{k,t+1}^{(d)}$ and $RV_{k,t}^{(d+w+m)}$ are $[1 \times n_k]$ and $[n_k \times 3]$ respectively, with n_k being the sample size of subgroup **k**. $||...||_q$ denotes the l_q norm of its argument and λ, γ, τ being the tuning parameters. Note that only τ (denoting the fusion-type penalty parameter) varies across the subgroups. This parameter is controlling to which extent similarities are encouraged between subgroup-specific coefficients. In equation (6) an l_2 fusion penalty can be seen, but like Dondelinger, Mukherjee, and Initiative (2020), also the use of an l_1 fusion penalty is considered.

The tuning parameters λ and γ control the amount of regularization, so choosing a good value of the tuning parameter seems crucial and therefore, these parameters are set by cross-validation. Each time the model is estimated, the data will be split into specified K folds. For each fold the data is divided into training data and validation data. For each tuning parameter value, the realized variances is estimated using the training data and the total error is recorded on the validation set of data. The tuning parameter for which the total error is smallest is chosen in the final model.

In contrast to the tuning parameters λ, γ, which are set by cross-validation, the parameter τ is computed otherwise. Dondelinger, Mukherjee, and Initiative (2020) give two approaches to set this parameter. For both approaches, $\tau_{k,k'}$ is calculated as follows:

$$\tau_{k,k'} = 1 - \frac{d(k,k')}{d_{max}} \tag{12}$$

Where d(k,k') is calculated in two ways.

Variant 1:

$$d(k,k') = ||\mu_k - \mu_{k'}||_2 \tag{13}$$

With μ_k, μ'_k are the sample means of the explanatory coefficients in the subgroups k and k'.

Variant 2:

$$d(k,k') = \frac{1}{2}(KL(\hat{p}_k||\hat{p}_{k'}) + KL(\hat{p}_{k'}||\hat{p}_k) \tag{14}$$

Where $\hat{p}_k, \hat{p}_{k'}$ are estimated distributions over the coefficients in the subgroups k, k'. $KL(p||q)$ is the Kullback-Leibler(KL)-divergence between distributions p and q. d_{max} denotes the largest value between any pair k and k'. Like stated in Dondelinger, Mukherjee, and Initiative (2020), the use of multivariate Normal models is appropriate for this purpose.

To obtain an optimal solution for equation (11), two methods are used as described in Dondelinger, Mukherjee, and Initiative (2020). The two methods are different for the type of l_q norm used for the fusion penalty. As described before, two variants for the fusion type penalty are considered, namely an l_1 and l_2 norm. The methods, as they are stated by Dondelinger, Mukherjee, and Initiative (2020), are displayed below.

l_2 fusion penalty: Coordinate descent

To solve the problem of equation (11) and properly optimize **B**, first the equation needs to be reformulated as a classical lasso problem seen in equation (10). If the formula in equation (11) is reformulated as a classical lasso problem, the GLMNET[1] software provided by Friedman, Hastie, and Rob Tibshirani (2010) can be used to obtain the values for B.

The first part of equation (11) and the third part are assembled into one. It is done as follows:

For simplicity denote $||RV^{(d)}_{k,t+1} - c - RV^{(d+w+m)}_{k,t}\beta_k||_2^2$ as $||y_{flat} - X_{diag} * b_{flat}||$, with X_{diag} a block-diagonal matrix: $X_{diag} = \begin{bmatrix} x_1 & & \\ & \ddots & \\ & & x_k \end{bmatrix}$ $b_{flat} = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_k \end{pmatrix}$ $y_{flat} = \begin{pmatrix} y_1 \\ \vdots \\ y_k \end{pmatrix}$

Now the l_2 fusion penalty $||\beta_k - \beta_{k'}||$ is moved into the first term by defining the augmented matrix X^{aug}_{diag} and augmented vector y^{aug}_{diag} , such that:

$$\widehat{b_{flat}} = \underset{b_{flat}}{argmin} ||y^{aug}_{diag} - X^{aug}_{diag} * b_{flat}||_2^2 + \lambda||b_{flat}||_1, \tag{15}$$

[1] https://cran.r-project.org/web/packages/glmnet/glmnet.pdf

where $X_{diag}^{aug} = \begin{pmatrix} X_{diag} \\ \Gamma \end{pmatrix}$ and $y_{diag}^{aug} = \begin{pmatrix} y_{flat} \\ \vec{0} \end{pmatrix}$

with Γ a matrix encoding the pair-wise fusion constraints, and $\vec{0}$ a vector of zeros. Each block $\Gamma_{k,k'}, k, k' \in [1, K], k < k'$ of p (number of explanatory variables) row of Γ corresponds to the fusion constraint between two coefficient vectors β_k and $\beta_{k'}$, with

$$\Gamma_{k,k}(l,m) = \begin{cases} \gamma\tau_{k,k'} & if \quad l = p(k-1)+m \\ -\gamma\tau_{k,k'} & if \quad l = p(k-1)+m \\ 0 & otherwise \end{cases}$$

A classical lasso problem can now be recognized in equation (11), to which GLMNET can be directly applied (Dondelinger, Mukherjee, and Initiative 2020).

l_1 fusion penalty: Proximal-gradient approach

An already discussed extension of the optimization problem in equation (11) is the use of an l_1 fusion penalty. It is possible to optimize equation (11) with l_1 fusion penalty, but it could be highly inefficient (Dondelinger, Mukherjee, and Initiative 2020). An optimization procedure based on a proximal gradient approximation is used and is derived in Chen et al. (2010). They describe a proximal relaxation of an optimization that introduces additional smoothing to turn the objective function $f_{L1}(B)$ (with l_1 norm) into a continuously differentiable function $f_{L2}^{\mu}(B)$. Dondelinger, Mukherjee, and Initiative (2020) adapted the proposed approach for a subgroup setting with different X_k for each subgroup. Such as done in the "Coordinate Descent" approach, they re-writed the problem first to contain only one penalty instead of two. To achieve this, first they introduce a graph formulation of the fusion penalties. For example, the undirected graph G = (V,E) with vertex set V = [1,...,k] corresponding to the each subgroup with edges between all vertices. For simplicity, again denote $||RV_{k,t+1}^{(d)} - c - RV_{k,t}^{(d+w+m)}\beta_k||_2^2$ as $||y_k - X_k * \beta_k||_2^2$. Then the l_1 penalised objective function can be written as:

$$f_{L1}(B) = \sum_k \left\{ \frac{1}{n_k} ||y_k - X_k * \beta_k||_2^2 \right\} + ||BC||_1, \tag{16}$$

where the last term includes both penalties via $C = (\lambda I_k, \gamma H)$ with I_k the identity matrix of size K and C a "$K \times |E|$" matrix, and where for any k $\in$ V and e = (m,l) $\in E$:

$$H_{k,e} = \begin{cases} \tau_{m,l} & if \quad k = m \\ -\tau_{m,l} & if \quad k = l \\ 0 & otherwise. \end{cases}$$

To optimize the equation above it is needed to calculate the gradient and have to make some minor modifications to replace $||BC||$. Like in Chen et al. (2010), the auxiliary matrix A $\in \zeta = A' | ||A'||_\infty \leq 1, A' \in R^{px(K+|E|)}$ Because of the duality between l_1 and l_∞, $||BC||_1 = max_{||A||_\infty \leq 1} \langle A, BC \rangle$. An approximation of $||BC||$

is then obtained by writing:

$$f_\mu(B) = max_{||A||_\infty \leq 1} \langle A, BC \rangle - \mu d(A), \tag{17}$$

where μ is a positive smoothness parameter and is set to $\mu = \frac{\epsilon}{p(K+|E|)}$, and d(A) = $\frac{1}{2}||A||_F^2$. Following Chen et al. (2010) the gradient of $f_\mu(B)$ is written as $\Delta f_\mu(B) = A^* C^T$, with A^* being the optimal solution of the equation above. Now $||BC||$ can be replaced by $f_\mu(B)$ and the next equations can be obtained:

$$\tilde{f}_{L1}(B) = \sum_k \left\{ \frac{1}{n_k} ||y_k - X_k * \beta_k||_2^2 \right\} + f_\mu(B), \tag{18}$$

with the gradient being denoted as:

$$\Delta \tilde{f}_{L1}(B) = \sum_k \left\{ \frac{1}{n_k} X_k^T (X_k * \beta_k - y_k) \right\} + f_\mu(B). \tag{19}$$

3.5 Cross-validation for tuning parameters

Getting an optimal value for equation (11) requires obtaining the values for the tuning parameters λ and γ. As stated before, cross validation is used for this purpose and this statistical method is explained in the next sentences. Firstly, the data is split randomly in a number of K groups, which are called folds. The choice for the number of folds is important as a poorly choice for may be mis-representative for the data sample. Typically, k = 5 or k = 10 used, as these values have been shown to yield neither high bias nor very high variance (James et al. 2013).

In this thesis most results are obtained using 5 folds, as it is more computationally efficient to do so. However, the difference in performance between 5 and 10 folds is shown in the result section. If the data is split into groups, then for each group the data is again split into training data and validation data. The training data consists of around 80% of the data points and the validation data consists of around 20% of the data points. The training data is used to fit the specified model and this model is tested and evaluated on the validation data. A performance measurement is used to rank all the tested models.

For each model that is tested a different value for λ and γ is used and the values with the best cross-validation score are saved. These values are later used in the final models that are used for one-step-ahead forecasts. The parameter values that are used in the cross-validation method are (0, 1e-8, 1e-7, 1e-6, 1e-5, 1e-4, 1e-3, 1e-2, 1e-1, 0.5, 1) for both λ and γ. These initial values used in the search for an optimal value are chosen randomly. A way to find initial values based on the data is by using the GLMNET package in R. However, better results are not obtained using this package and more importantly it is more computationally efficient to use the values above.

3.6 Forecasting with time series models

After the creation of the Joint Lasso model as described in the sections above, out-of-sample predictions can be made. The predictive regression model is used, which Rapach, Strauss, and Zhou (2010) describe as:

$$r_{t+1} = \alpha + \theta_t x_t + \epsilon_{t+1} \tag{20}$$

where in the setup of this thesis r_{t+1}, the dependent variable, is denoted as the realized volatility at time t+1. The coefficients θ are in this case time-varying and a constant is included in the regression model. In this thesis the coefficient θ is estimated with OLS and with the Joint Lasso. The equation can be written for this purpose as:

$$RV_{t+1} = c + \beta_t RV_t + \omega_{t+1} \tag{21}$$

Out-of-sample analysis is performed for both estimation methods and one can choose between a moving window or extending window to do so. A moving window is used in this thesis and the sizes differ between 50 observations and 200 observations. The outcomes of the different estimation windows can be found in the results section.

To clarify, when performing one-step-ahead forecasts, one estimates a model which uses data from a certain period, the estimation period. The coefficients of the model are used to predict an observation that is the next observation outside the estimation period (or the first observation of the forecast period). Then a new model is estimated using the observation that was forecasted in the last step and deleting the first observation of the estimation period. Another prediction is made for the next observation (the second observation of the forecast period) and the steps are repeated.

3.7 Performance measurement

For statistical evaluation of the forecasted realized volatilities, the root mean squared error (RMSE) is used, as in Čech and Barunık (2017). The RMSE is defined as follows:

$$RMSE = \sqrt{\sum_{t=t+h}^{T} (e_t)^2} \tag{22}$$

$$e_t = \widehat{RV}_t - RV_t, \tag{23}$$

where $\widehat{RV}$ is the forecasted realized volatility and RV_t the actual value of the realized volatility. The forecast sample is between (t+h) and T.

4 Data

High-frequency daily volatility data collected by Wharton Research Data Services (WRDS) is used from Wharton University of Pennsylvania.[2]

The database of WRDS contains intraday transactions data (trades and quotes) for all securities listed on the New York Stock Exchange (NYSE) and American Stock Exchange (AMEX), as well as Nasdaq National Market System (NMS) and SmallCap issues.

The data needed for the research is processed as described in the methodology section. As mentioned, intraday Trade-based and Quote-based high-frequency data is used for the individual stocks listed on the

[2] https://wrds-www.wharton.upenn.edu

NYSE. The data set used in this thesis is provided by Noureldin, Shephard, and Sheppard (2012) and is slightly adapted for the purposes of this thesis. The data set contains daily realized variances and covariances for 10 DJIA stocks. In computing the realized covariance matrix, 5-minute returns with subsampling are used, so the resulting subsampled realized covariance is much more robust to the so-called market microstructure noise than the simple 5-minute based one (Chiriac and Voev 2011). The first column of the data set contains the dates with the other columns containing the calculated volatilities. The dates vary from the first of February 2001 until the 31st of December 2009. As the model is a time series model with a lag structure, only the data from the 6th of March 2001 until the 31st of December 2009 is used.

The 55 columns are in the form of the vech of the $[10 \times 10]$ realized covariance matrix where the stocks are ordered as follows: Bank of America (BAC), JP Morgan (JPM), International Business Machines (IBM), Microsoft (MSFT), Exxon Mobil (XOM), Alcoa (AA), American Express (AXP), Du Pont (DD), General Electric (GE) and Coca Cola (KO). Applying an inverse vech operator gives the realized covariance matrix. The realized volatilities that are used are stated in the columns (1,11,20,28,35,41,46,50,53,55). The lagged realized volatilities needed for the HAR-model can be easily calculated.

4.1 cleaning the data

Barndorff-Nielsen, P Reinhard Hansen, et al. (2009) state the importance of cleaning the data when working with Trade and Quote data. They show that biases and inconsistencies may arise and recommend the steps below before applying any volatility analysis. In their procedure, Barndorff-Nielsen, P Reinhard Hansen, et al. (2009) differentiate between trade data or quote data. The data set used in this thesis is cleaned according to the following procedure:

1. Restrict data to exchange hours
2. Delete entries with a transaction price equal to zero
3. Restrict data to specific exchange
4. Delete entries with corrected trades (where CORR = 0)
5. Delete entries with an abnormal sale condition (Trades with letter code except for 'E' and 'F')
6. If multiple transactions have the same time stamp, use the median price
7. Delete entries for which the price deviated by more than 10 mean absolute deviations from a rolling centered median (excluding the observation under consideration) of 50 observation (25 observations before and 25 after)
8. Delete entries with zero quotes
9. Delete entries with same time stamp and use median quotes
10. Delete entries with negative spreads

4.2 summary statistics

Table 1: Summary statistics for set of stocks between 2001-2009

Summary statistics

Company name	**Ticker**	**sector**	**Avg. daily Volatility**	**Max. volatility**
Bank of America	(BAC)	Finance	5.457	277.307
JP Morgan	(JPM)	Finance	5.055	176.478
International Business Machines	(IBM)	Technology	1.933	57.542
Microsoft	(MSFT)	Technology	2.454	43.105
Exxon Mobil	(XOM)	Energy	2.072	115.378
Alcoa	(AA)	Manufacturing	4.943	160.241
American Express	(AXP)	Finance	4.419	201.878
Du Pont	(DD)	Manufacturing	2.528	63.873
General Electric	(GE)	Energy	3.195	114.255
Coca Cola	(KO)	Manufacturing	1.414	56.505

Table 1 show the statistics of the set of stocks. It is seen that the maximum volatility is really high, compared to the average volatility. The maximum volatilities are all taking place in 2008, during the economic crisis. In figure 1 the course of the realized volatility is shown for the stock BAC. Between 2001 and 2007 the maximum volatility is not above 20, while in the periode between 2008 and 2009 the volatility is on average around 20. Thus, the model created can be tested on several economic periods. The graphs for the other stocks can be found in the Appendix.

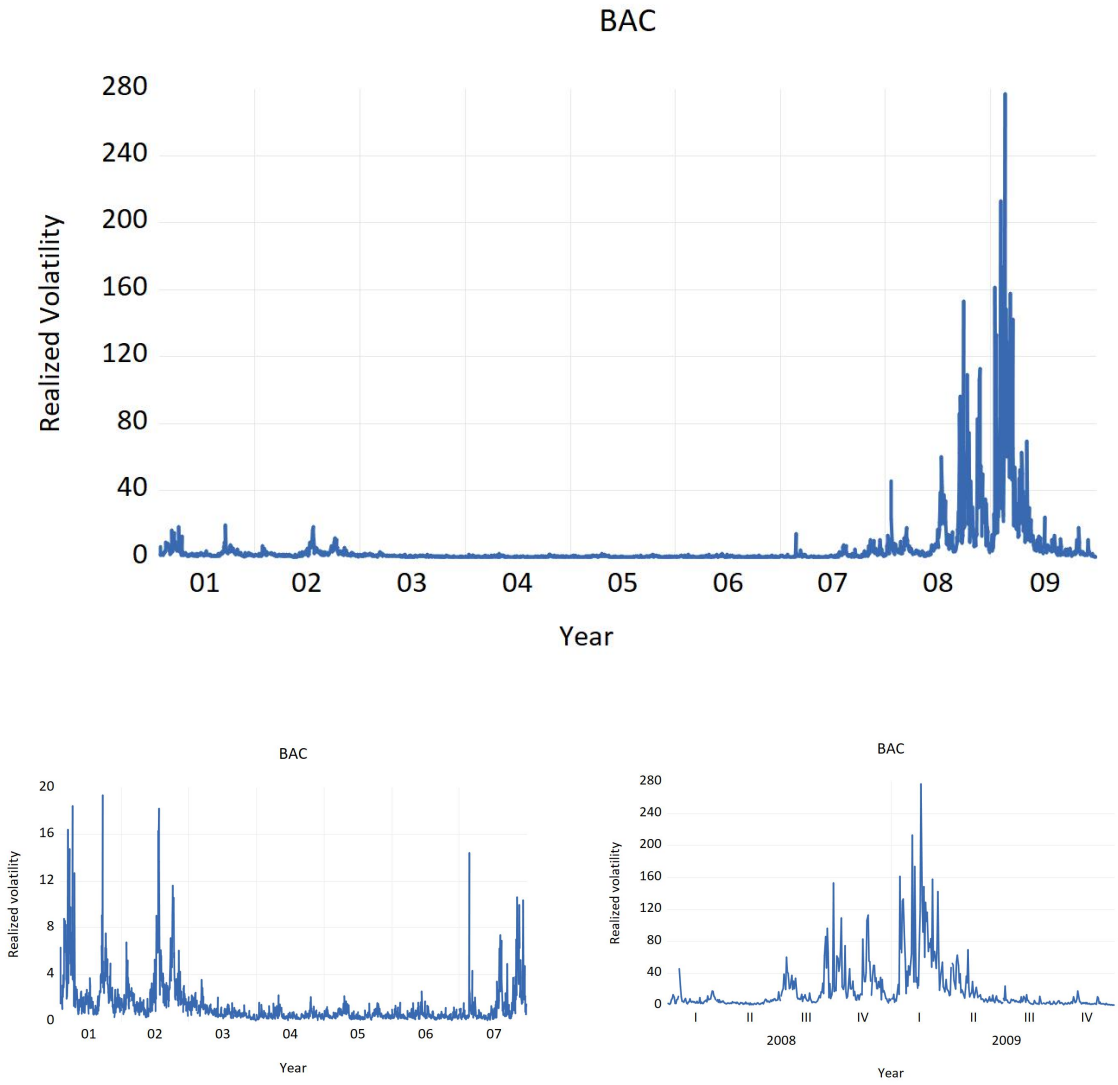

Figure 1: Realized volatilities for different periods for BAC

5 Results

The result section shows the performance of the different models that are constructed. The performance measure that is used is the "root mean squared error" (RMSE). The performance of the constructed models is compared with a so called benchmark model, which is in this cas the equation-by-equation (for each diagonal element of the covariance matrix) OLS-model. As described in the previous sections, the sample that is used dates from the 6th of March 2001 until the 31st of December 2009 containing 2220 observations. If OLS is performed on the whole sample, the following model is obtained:

Table 2: Ordinary Least Squares coefficients of the 10 assets for the whole sample

OLS coefficients

Company name	**Coefficients**			
	Constant	RV_{t-1}	RV_{t-5}	RV_{t-22}
Bank of America	0.370	0.320	0.338	0.273
JP Morgan	0.505	0.559	0.060	0.279
International Business Machines	0.162	0.074	0.644	0.189
Microsoft	0.179	0.311	0.455	0.154
Exxon Mobil	0.257	0.073	0.745	0.058
Alcoa	0.380	0.266	0.519	0.138
American Express	0.364	0.125	0.534	0.256
Du Pont	0.219	0.199	0.582	0.131
General Electric	0.330	0.332	0.406	0.157
Coca Cola	0.146	0.149	0.621	0.123

As seen in table 2, the underlying models are different for each assets. However, some similarities can be found and groups can be formed accordingly. The table shows that Microsoft and General Electric have similar coefficients, for example. Another way of grouping the assets is by sector as mentioned before. Next to grouping by sector, Kumar (2017) also investigates grouping assets by size. Unfortunately the companies used in this thesis all have a big market gap, so only grouping by sector will be possible. A last way of grouping is to group the assets randomly, to see what way of grouping is optimal. The groups that are formed are the following:

Table 3: The table shows to wich group each company belongs. The groups are indicated with letters.

Assets per group

Company name	**Grouped by:**			
	Company	**Sector**	**OLS Coefficients**	**Random**
Bank of America	a	a	a	a
JP Morgan	b	a	b	a
International Business Machines	c	b	c	b
Microsoft	d	b	a	c
Exxon Mobil	e	c	d	d
Alcoa	f	d	e	e
American Express	g	a	c	e
Du Pont	h	d	e	b
General Electric	i	c	a	a
Coca Cola	j	d	c	b
Number of groups:	10	4	5	5

The models compared are the Joint Lasso models where the data is grouped in a certain way. The differences present in the Joint Lasso is the type of fusion penalty used. The l_2 approach leverages the GLMNET package and it is more computationally efficient than the l_1 approach (Dondelinger, Mukherjee, and Initiative 2020). This is the reason why most results are obtained using the l_2 approach. The data is grouped by company, sector, random or by coefficients, as stated in table 3. Two models are compared with a varying rolling window size between 50 and 200 observations. The following table shows the results, when the assets are put in groups based on the sector, coefficients or put together randomly as mentioned in table 3. Table 1 denotes the sector for each asset for reference.

Table 4: Statistical results of the predicting models in the full sample. Two rolling windows are used and different models are compared with some minor extensions per model. The equation-by-equation OLS is defined as OLS and the joint lasso is defined by the type of fusion penalty, L1 or L2.

Predicting models, full sample

		Window size = 200	**Window size = 50**	
Method	Grouped	RMSE	RMSE	Difference
OLS		6.165	6.553	0.389
L2	Company	5.825	6.202	0.377
L2	Sector	6.449	6.278	−0.171
L1	Sector	6.355	6.260	−0.094
L2	Random	6.434	6.446	0.012
L2	Coefficients	6.358	6.403	0.045

Looking at table 4, all the results for the different models are displayed. All the constructed models have a RMSE of around 6. It is seen that a decrease in the rolling window size from 200 to 50 results in an increase in the RMSE for the OLS model. A simple explanation for this result could be the fact that a larger rolling window size means more information that can be used to make more accurate predictions. The OLS model records the highest RMSE and the biggest difference between the two window sizes. This is in line with the hypothesis, as all the Joint Lasso models use different forms of information sharing to make the predictions more accurate. The model with the lowest RMSE is the Joint Lasso model where groups are formed based on the company. As mentioned before, in this model each asset is put in a separate group. This means that for each asset different coefficients are estimated, so dissimilarities between the assets do not lead to inaccuracies. Similarities between the assets will however, lead to more accurate predictions due to the information sharing nature of the Joint Lasso. This could be an explanation why the L2-model with grouping based on company has the lowest RMSE. The performance of this model decreases when the window size decreases, just like the OLS-model.
Only the Joint Lasso models with l_1 and l_2 fusion penalty result in a decrease of the RMSE when the window size decreases. This decrease of the RMSE is not seen with the l_2 models that are grouped randomly or based on the OLS coefficients. A reason for this may be that the groups are not formed optimal. This can be argued by all means for the randomly formed group. Nevertheless, the difference between the RMSE using different rolling window sizes is smaller than the OLS model. It should be noted that the number of groups of the models grouped by sector is 4, while the number of groups of the models where groups are formed randomly or based on coefficients is 5. A decrease in the number of groups results in an increase in the number of observations that are used for estimating the group coefficients. Especially when the window size is limited to 50, less groups formed can improve accuracy.

Table 5: Statistical results of the predicting models in the full sample. The Joint Lasso is used for both models with an l2 fusion penalty. The assets are grouped by sector in both models.

Different folds

		Window size = 200	**Window size = 50**	
Method	#folds	RMSE	RMSE	Difference
L2	5 folds	6.449	6.278	−0.171
L2	10 folds	6.409	6.288	−0.120

In table 5, the difference in performance can be seen when 10 folds or 5 folds are used in the cross-validation process. The advantage of using 10 folds instead of 5 folds is the improved accuracy of the choice for the tuning parameters. Choosing a tuning parameter that is less appropriate could have big consequences, especially because the difference between the initial values for λ and γ is big. An advantage of 5-fold cross-validation is that this method is computationally more efficient. That is why for all results in table 4 only results obtained from 5-fold cross-validation are noted.

Table 6: Statistical results of the predicting models in the full sample. The results are showed for two models where one model uses scaled data and the other uses non-scaled data.

Scaled data grouped by sector

		Window size = 200	**Window size = 50**	
Method	Scaled	RMSE	RMSE	Difference
L1	Scaled	6.720	7.177	0.456
L1	not-scaled	6.355	6.260	−0.094

Table 6 shows that scaling the data before estimating the model does not result in better performance. For this reason only results are obtained where the data is not scaled.

Table 7: Statistical results of the predicting models in the sample between 2008 and 2009. Two rolling windows are used and several models are compared. The equation-by-equation OLS is defined as OLS and the joint lasso is defined by the type of fusion penalty, L1 or L2.

Sample period 2008-2009

		Window size = 200	**Window size = 50**	
Method	Grouped	RMSE	RMSE	Difference
OLS		13.652	12.288	−1.364
L2	Company	13.309	11.942	−1.367
L2	Sector	14.768	12.662	−2.107
L1	Sector	14.344	13.131	−1.212

Table 7 shows results for the different models in the sample period between 2008 and 2009. This sample period is considered as an economic turbulent time. All the maximum realized volatilites are also taking place in this sample period. It is not remarkable that the RMSE for each model is higher than in table 4. It is also seen that all models in the table have a better performance when a rolling window size of 50 is used. Even the OLS-model is performing better when a smaller rolling window is used, in contrast to the results in table 4. A conclusion that can be drawn from this table is that the use of more recent information outweighs the amount of information in economic turbulent times.

Table 8: Statistical results of the predicting models in the sample between 2001 and 2007. Two rolling windows are used and several models are compared. The equation-by-equation OLS is defined as OLS and the joint lasso is defined by the type of fusion penalty, L1 or L2.

Sample period 2001-2007

		Window size = 200	**Window size = 50**	
Method	Grouped	RMSE	RMSE	Difference
OLS		2.400	3.189	0.788
L2	Company	1.870	2.599	0.729
L2	Sector	2.371	2.454	0.084
L1	Sector	2.273	2.471	0.198

Table 8 shows similar results compared to table 4. It is seen that the OLS model has the highest RMSE and it increases the most when a smaller rolling window size is used. In contrast to the the full sample, this sample shows no improvements in the performance of the models when a smaller window size is used. The L2-model where groups are formed based on the sector outperforms the model where each of the 10 assets is put in a separate group if a smaller window size is used. Thus, grouping based on sector is more consistent and robust than grouping based on the company.

6 Conclusion

To state the main research question of this thesis again: "Can the predictive performance of the multivariate HAR model for diagonal elements of the covariance matrix be improved with the Joint Lasso?". In line with the hypothesis, the predictive power of the HAR model can be improved when making use of information sharing between different assets. The results are even more pronounced if a small forecast window is used, as the Joint Lasso makes use of common information between the assets to make accurate forecasts. All in all, the implementation of models that make use of a joint information structure in modelling realized volatilies looks promising.

A sub-question this thesis tries to answer is: "Can the rolling window size be reduced using the Joint Lasso to improve the performance in economic turbulent times?" The use of the Joint Lasso is investigated in several sample periods and it is concluded that when the full sample is used the model always outperforms equation-by-equation OLS. However, for the sample period 2008-2009, the Joint Lasso does not necessarily outperforms the OLS model. Independent of this, the Joint Lasso is more robust to changes of the rolling window size.

Furthermore, the way of grouping the assets is investigated. Grouping based on sector is compared to grouping randomly. Grouping based on sector seems justifiable based on the results, as the perfor-

mance increases when a smaller rolling window size is used. This is a really important feature, as it is shown that a smaller rolling window size is always beneficial when constructing a model for economic turbulent times. Next to the formation of groups, the difference in the number of folds used in the cross-validation process is discussed. Corresponding to the hypothesis and the literature, 10-fold cross-validation outperforms 5-fold cross-validation, although the difference is minimal. Lastly, the influence of scaling the data is investigated. No gains can be made by scaling the data.

Further studies to extend this research can be devoted to the optimization of the cross-validation process. For example, a small set of initial values is used in the process and these values are selected almost randomly. A method to optimize the choice for the initial tuning parameters can be used. Another point of interest could be the use of the Kullback-Leibler(KL)-divergence (as mentioned in the methodology) for setting the tuning parameter τ. Unfortunately this method turned out to be computationally inefficient, and this is the reason it is not investigated in this thesis.

7 Appendix

7.1 Realized volatility graphs for all 10 assets

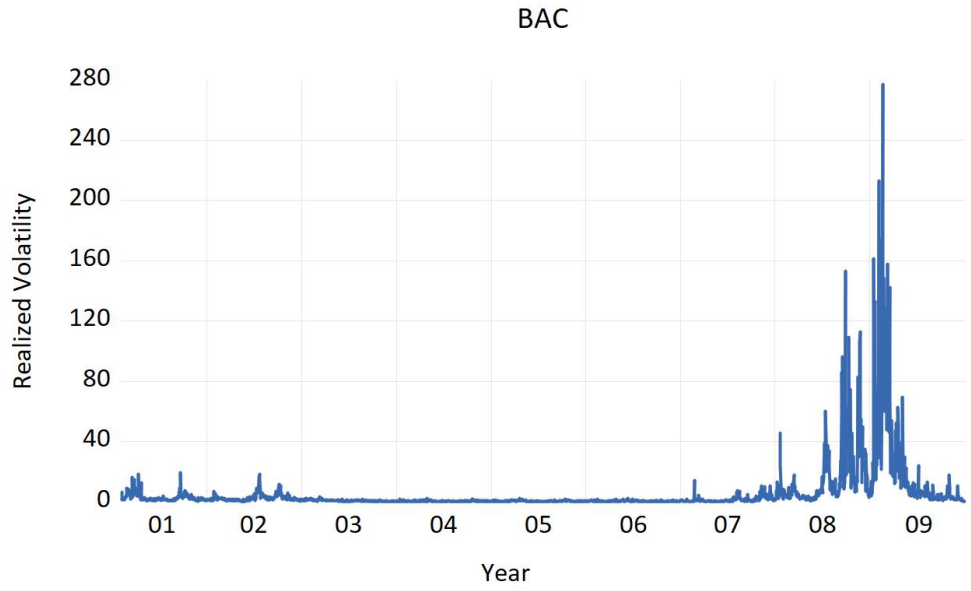

Figure 2: Realized volatility 2001M01-2009M12

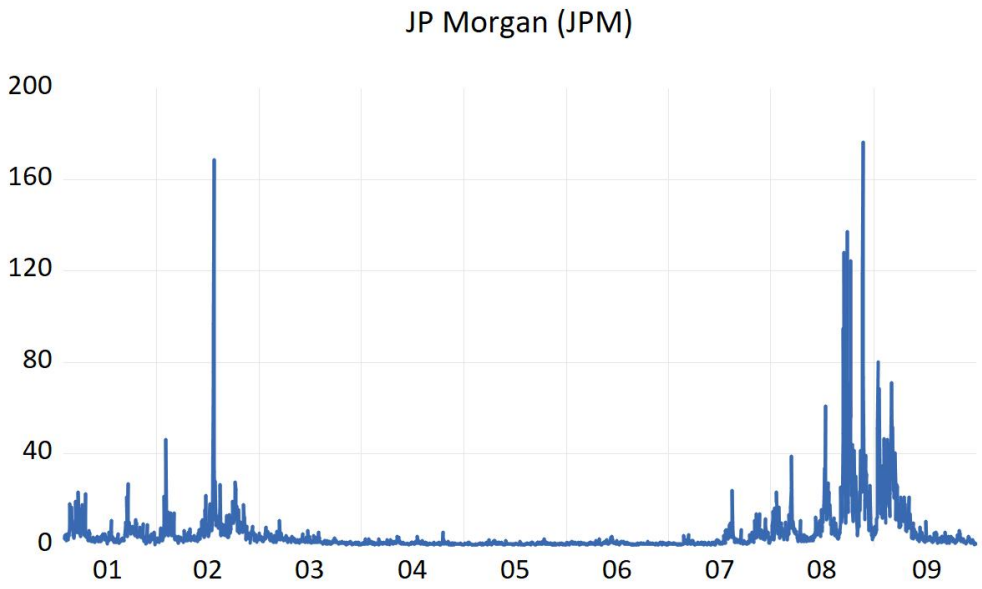

Figure 3: Realized volatility 2001M01-2009M12

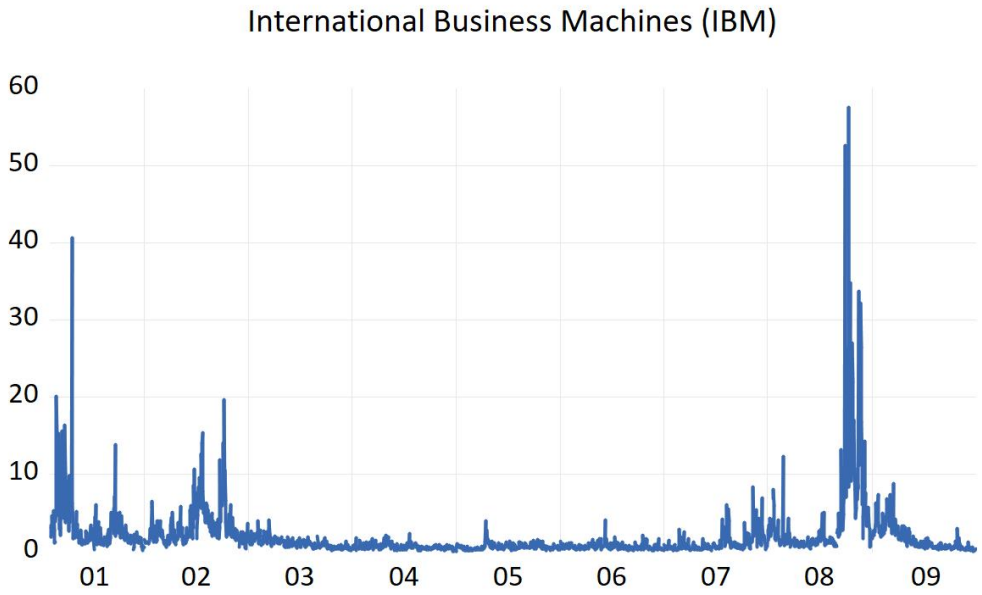

Figure 4: Realized volatility 2001M01-2009M12

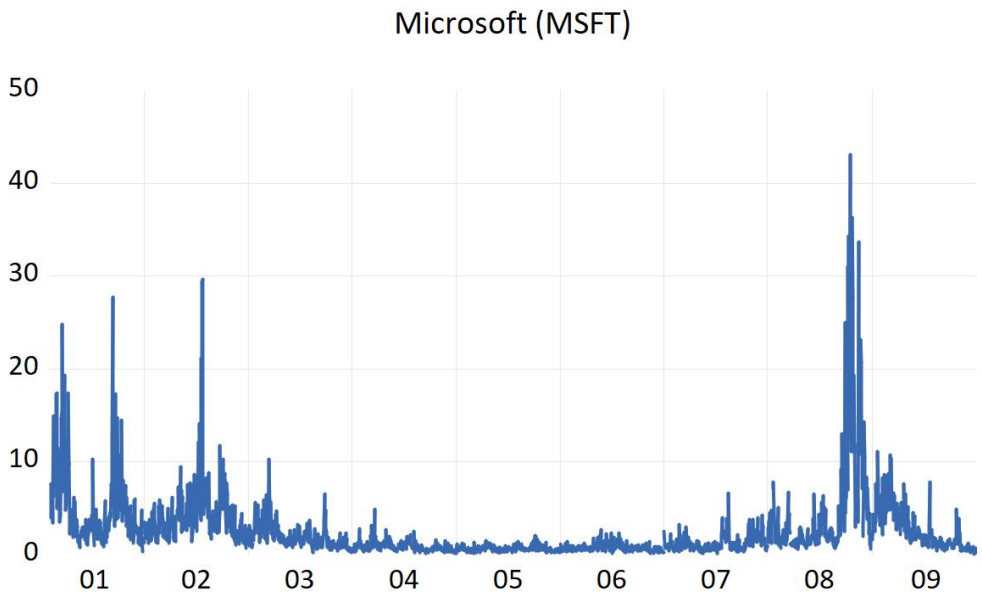

Figure 5: Realized volatility 2001M01-2009M12

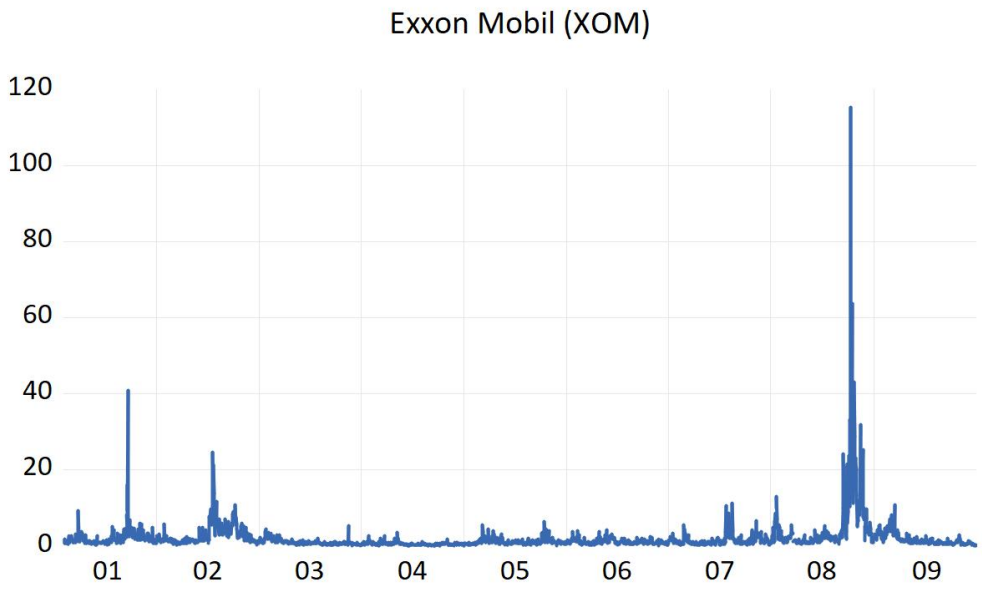

Figure 6: Realized volatility 2001M01-2009M12

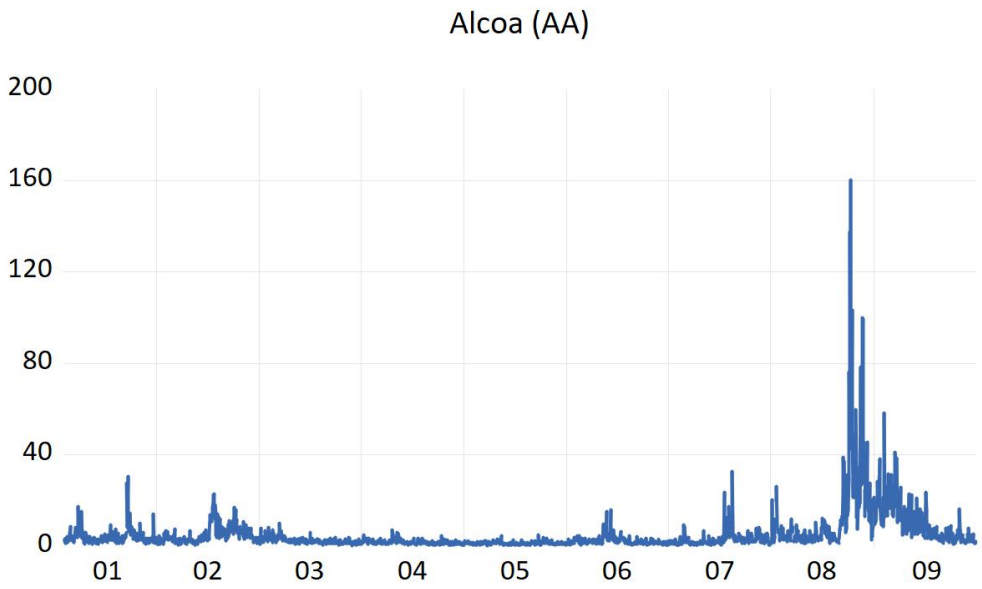

Figure 7: Realized volatility 2001M01-2009M12

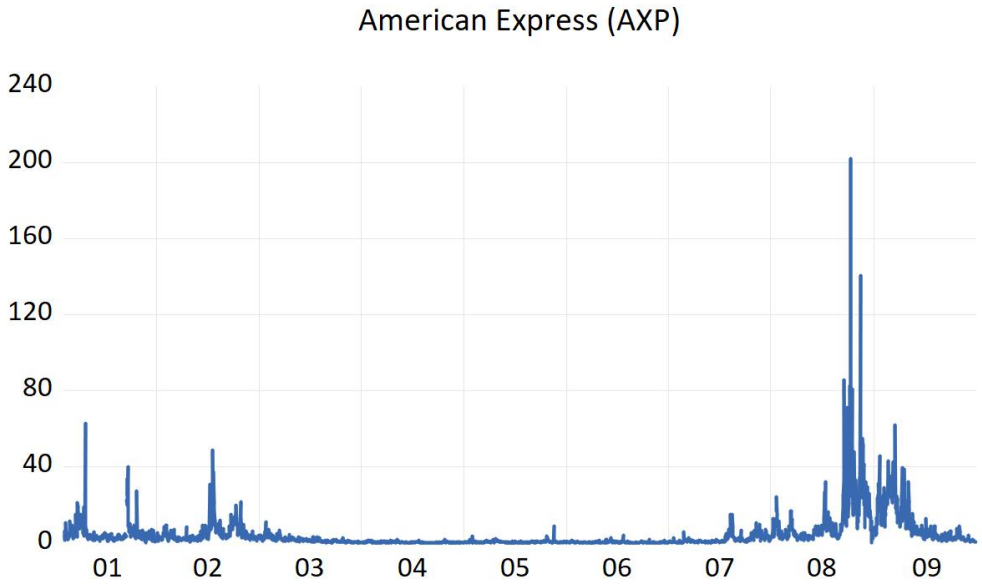

Figure 8: Realized volatility 2001M01-2009M12

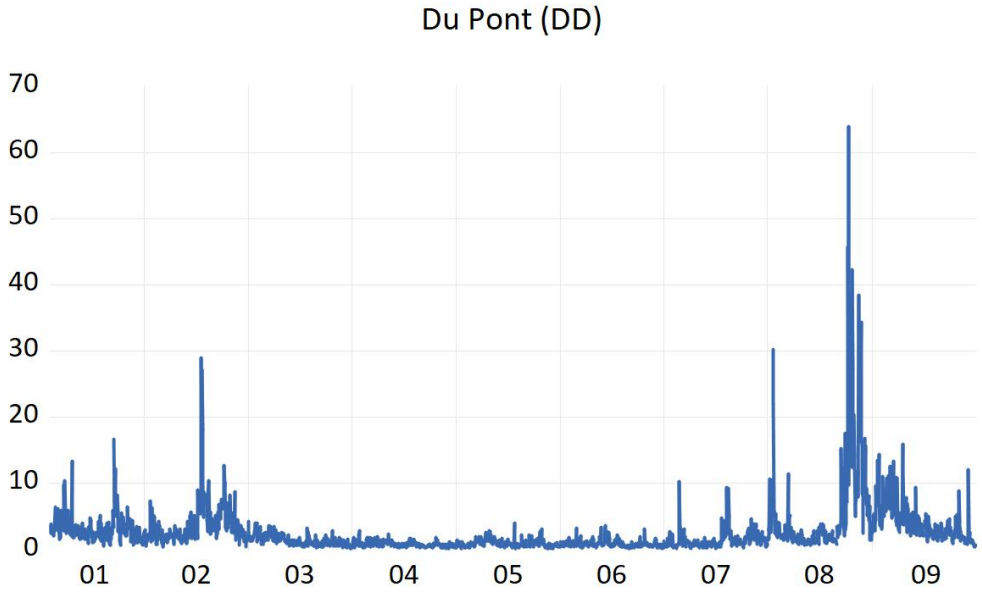

Figure 9: Realized volatility 2001M01-2009M12

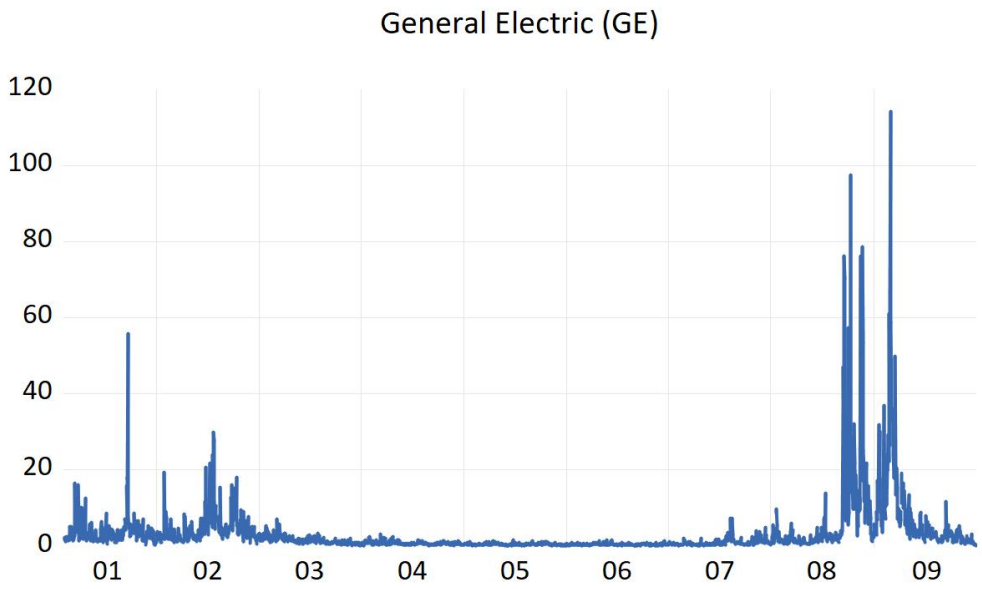

Figure 10: Realized volatility 2001M01-2009M12

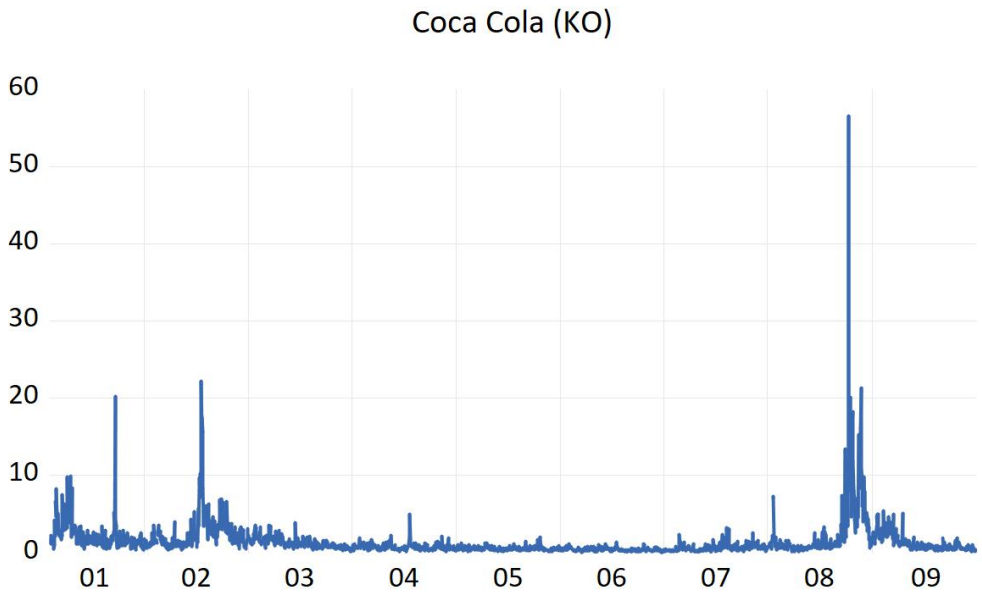

Figure 11: Realized volatility 2001M01-2009M12

8 References

[1] Torben G Andersen, Tim Bollerslev, and Nour Meddahi. "Realized volatility forecasting and market microstructure noise". In: *Journal of Econometrics* 160.1 (2011), pp. 220–234.

[2] Francesco Audrino and Lorenzo Camponovo. "Oracle properties and finite sample inference of the adaptive lasso for time series regression models". In: *arXiv preprint arXiv:1312.1473* (2013).

[3] Francesco Audrino, Chen Huang, and Ostap Okhrin. "Flexible HAR model for realized volatility". In: *Studies in Nonlinear Dynamics & Econometrics* 23.3 (2018).

[4] Francesco Audrino and Simon D Knaus. "Lassoing the HAR model: A model selection perspective on realized volatility dynamics". In: *Econometric Reviews* 35.8-10 (2016), pp. 1485–1521.

[5] Ole E Barndorff-Nielsen, P Reinhard Hansen, Asger Lunde, and Neil Shephard. *Realized kernels in practice: Trades and quotes*. 2009.

[6] Ole E Barndorff-Nielsen, Peter Reinhard Hansen, Asger Lunde, and Neil Shephard. "Designing realized kernels to measure the ex post variation of equity prices in the presence of noise". In: *Econometrica* 76.6 (2008), pp. 1481–1536.

[7] Ole E Barndorff-Nielsen and Neil Shephard. "Econometric analysis of realized volatility and its use in estimating stochastic volatility models". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 64.2 (2002), pp. 253–280.

[8] František Čech and Jozef Baruník. "On the modelling and forecasting of multivariate realized volatility: Generalized heterogeneous autoregressive (GHAR) model". In: *Journal of forecasting* 36.2 (2017), pp. 181–206.

[9] Xi Chen, Seyoung Kim, Qihang Lin, Jaime G Carbonell, and Eric P Xing. "Graph-structured multi-task regression and an efficient optimization method for general fused lasso". In: *arXiv preprint arXiv:1005.3579* (2010).

[10] Roxana Chiriac and Valeri Voev. "Modelling and forecasting multivariate realized volatility". In: *Journal of Applied Econometrics* 26.6 (2011), pp. 922–947.

[11] Fulvio Corsi. "A simple approximate long-memory model of realized volatility". In: *Journal of Financial Econometrics* 7.2 (2009), pp. 174–196.

[12] Fulvio Corsi, Francesco Audrino, and Roberto Renó. "HAR modeling for realized volatility forecasting". In: (2012).

[13] Frank Dondelinger, Sach Mukherjee, and Alzheimer's Disease Neuroimaging Initiative. "The joint lasso: high-dimensional regression for group structured data". In: *Biostatistics* 21.2 (2020), pp. 219–235.

[14] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. "Regularization paths for generalized linear models via coordinate descent". In: *Journal of statistical software* 33.1 (2010), p. 1.

[15] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*. Vol. 112. Springer, 2013.

[16] Dilip Kumar. "Realized volatility transmission from crude oil to equity sectors: A study with economic significance analysis". In: *International Review of Economics & Finance* 49 (2017), pp. 149–167.

[17] Diaa Noureldin, Neil Shephard, and Kevin Sheppard. "Multivariate high-frequency-based volatility (HEAVY) models". In: *Journal of Applied Econometrics* 27.6 (2012), pp. 907–933.

[18] David E Rapach, Jack K Strauss, and Guofu Zhou. "Out-of-sample equity premium prediction: Combination forecasts and links to the real economy". In: *The Review of Financial Studies* 23.2 (2010), pp. 821–862.

[19] Bonnie K Ray and Ruey S Tsay. "Long-range dependence in daily stock volatilities". In: *Journal of Business & Economic Statistics* 18.2 (2000), pp. 254–262.

[20] Ghulam Sarwar. "The interrelation of price volatility and trading volume of currency options". In: *Journal of Futures Markets: Futures, Options, and Other Derivative Products* 23.7 (2003), pp. 681–700.

[21] Robert Tibshirani. "Regression shrinkage and selection via the lasso". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1 (1996), pp. 267–288.

[22] Dimitrios I Vortelinos. "Forecasting realized volatility: HAR against Principal Components Combining, neural networks and GARCH". In: *Research in international business and finance* 39 (2017), pp. 824–839.

[23] Lan Zhang, Per A Mykland, and Yacine Aıt-Sahalia. "A tale of two time scales: Determining integrated volatility with noisy high-frequency data". In: *Journal of the American Statistical Association* 100.472 (2005), pp. 1394–1411.

[24] Hui Zou. "The adaptive lasso and its oracle properties". In: *Journal of the American statistical association* 101.476 (2006), pp. 1418–1429.