

Realized Rotated Multivariate GARCH Model¹

Denis Pelletier

North Carolina State University²

and

Ji Shen

North Carolina State University³

May 21, 2015

¹xxxxxxx.

²Department of Economics, Box 8110, North Carolina State University, Raleigh, NC 27695-8110, USA. Email:
denis.pelletier@ncsu.edu

³xxxxxxx

Abstract

xxxxxx ?xxxxxxxxxx.

JEL classification: C10, C32

Keywords: xxxxxx, xxxxxx, xxxxxx.

1 Introduction

2 Realized Multivariate GARCH model

2.1 Definitions and Notations

In this section we introduce the Realized multivariate GARCH model. Assuming we have k assets, r_t is the $k \times 1$ vector of returns for the k assets in day t . We will assume $E(r_t|\mathcal{F}_{t-1}) = 0$, which is a legitimate assumption from the empirical work. Otherwise we could always reinterpret r_t as the return minus its constant mean to meet this assumption. The key interest of financial application resides on the conditional variance, $H_t = \text{var}(r_t|\mathcal{F}_{t-1})$, where r_t is a vector of time series of returns, \mathcal{F}_{t-1} is the filtration till time $t-1$. In the GARCH (1, 1) model the conditional variance is a function of lagged itself and squared daily return. In the present framework, H_t will depend on X_{t-1} , which represents a realized measure of volatility, such as matrix of realized covariance, bipower variation or realized kernel. The measurement equation link the realized measure to the latent make the model complete. Thus the Realized GARCH model fully specifies the dynamic system of daily returns and realized measure.

The main challenges in multivariate volatility modelling is to ensure that the conditional covariance matrix is positive semidefinite. The BEKK model proposed by Engle and Kroner(1995) could approach this challenge . We could specify a particular BEKK-type system as follows:

$$\begin{aligned} r_t &= H_t^{1/2} z_t \\ z_t &\stackrel{i.i.d}{\sim} \mathcal{N}(0, I_k) \\ E[P_t|\mathcal{F}_{t-1}] &= E[r_t r_t'|\mathcal{F}_{t-1}] = H_t \\ P_t &= H_t^{1/2} \xi_t H_t^{1/2} \end{aligned} \tag{1}$$

where conditional covariance follows a GARCH(1,1) process

$$H_t = CC' + B H_{t-1} B + A X_{t-1} A \tag{2}$$

and

$$x_t = D + \Gamma h_t + u_t \tag{3}$$

We refer to first two equation as the return equation and the Realized GARCH equation. Since the realized measure X_t could be interpreted as a measurement of conditional covariance H_t , the last

equation is referred as the measurement equation. In (1), $\xi_t = z_t z_t'$ is $k \times k$ symmetric innovation matrix which is outer product of innovation vector z_t , satisfying $E_{t-1}[\xi_t] = I_k$, where I_k is an identity matrix. In (2), we define the symmetric square root of a positive semidefinite matrix H , denoted by $H^{1/2}$, using the spectral decomposition such that $H^{1/2} = U \Lambda^{1/2} U'$ where U is a matrix containing the eigenvectors of H , and $\Lambda^{1/2}$ is a diagonal matrix containing the square root of the eigenvalues of H . In (3) We define $x_t = \text{vech}(\log X_t)$ and similarly $h_t = \text{vech}(\log H_t)$ where the vech operator stacks the lower triangular part including the main diagonal of a $(k \otimes k)$ symmetric matrix into a $(k^* \otimes 1)$ column vector, where $k^* = k(k+1)/2$. We use \log before matrix to represent the principal matrix logarithm. For most matrices, $\log(\exp(A)) = A = \exp(\log(A))$.

2.2 Parameterization

The $k \times k$ matrices B , A each have k^2 free parameters, while C is $k \times k$ a lower triangular matrix with k^* free parameters. The parameterization in (2) guarantee that H_t is positive semidefinite for all t assuming the starting value H_0 is positive semidefinite. Moreover, if C is full rank matrix, then H_t is positive definite for all t . The coefficient matrix B determines the persistence of the Realized MGARCH model. To make sure the covariance stationarity, the eigenvalues of the coefficient matrix B must be less than one in modulus. Since X_t and H_t are supposed to be symmetric matrix, the vech operator vectorizing the low triangular part including the main diagonal of a $k \times k$ symmetric matrix into a $k^* \times 1$ vector. Hence, x_t , D , h_t and u_t are all $k^* \times 1$ vectors. Moreover, the innovation $u_t \stackrel{i.i.d}{\sim} \mathcal{N}(\mu, \delta^2)$. The coefficient matrix Γ is a $k^* \times k^*$ matrix. This matrix characterize the correlation between realized measure and all the elements of conditional covariance matrix. Specifically it forms a seemingly unrelated regression between the two variables. As a result, the equation (3) will have $O(k^4)$ parameters, which will suffer the curse-of-dimensionality problem. Since the measurement equation could be estimated with linear regression, which is an easy task in computer. So the problem of high dimensionality is not about burden of computation but the prediction variation produced by too many parameters.

Even the BEKK-type parameterization in (2) has $O(k^2)$ parameters. To avoid the curse-of-dimensionality problem here we could impose that A and B are diagonal matrices, which yields the diagonal Realized MGARCH model. In this case, the equations for each element in the matrix H_t would characterize a bunch of univariate Realized GARCH models in which the conditional variance or covariance is driven by their own lags and the corresponding realized measure. If the elements of A , B are unrestricted which

is a full parameterization, the evolution of every element in H_t will be influenced by their own lags as well as every cross-asset effects Realized MGARCH model.

For demonstration, we write out the 2 by 2 case of (2)

$$\begin{pmatrix} h_{11,t} & h_{12,t} \\ h_{21,t} & h_{22,t} \end{pmatrix} = \begin{pmatrix} c_{11} & 0 \\ c_{21} & c_{22} \end{pmatrix} \begin{pmatrix} c_{11} & 0 \\ c_{21} & c_{22} \end{pmatrix}' \\ + \begin{pmatrix} b_{11} & 0 \\ 0 & b_{22} \end{pmatrix} \begin{pmatrix} h_{11,t-1} & h_{12,t-1} \\ h_{21,t-1} & h_{22,t-1} \end{pmatrix} \begin{pmatrix} b_{11} & 0 \\ 0 & b_{22} \end{pmatrix} \\ + \begin{pmatrix} a_{11} & 0 \\ 0 & a_{22} \end{pmatrix} \begin{pmatrix} x_{11,t-1} & x_{12,t-1} \\ x_{21,t-1} & x_{22,t-1} \end{pmatrix} \begin{pmatrix} a_{11} & 0 \\ 0 & a_{22} \end{pmatrix}$$

We could write out (3) of 2 by 2 case in vec form

$$\begin{pmatrix} h_{11,t} \\ h_{21,t} \\ h_{22,t} \end{pmatrix} = \begin{pmatrix} d_{11,t} \\ d_{21,t} \\ d_{31,t} \end{pmatrix} + \begin{pmatrix} \gamma_{11,t} & \gamma_{12,t} & \gamma_{13,t} \\ \gamma_{21,t} & \gamma_{22,t} & \gamma_{23,t} \\ \gamma_{31,t} & \gamma_{32,t} & \gamma_{33,t} \end{pmatrix} \begin{pmatrix} h_{11,t} \\ h_{21,t} \\ h_{31,t} \end{pmatrix} + \begin{pmatrix} u_{11,t} \\ u_{21,t} \\ u_{31,t} \end{pmatrix}$$

We could tell from the example that in the measurement equation, each element in the conditional covariance matrix in function of every component in the realized covariance. Clearly the measurement equation is over-parameterized. We could use shrinkage and variable selection method in the estimation to reduce elements in coefficient matrix Γ .

Note that, the (2) could easily written in its vec form

$$h_t = \bar{C} + \bar{B} h_{t-1} + \bar{A} x_{t-1} \quad (4)$$

Which $\bar{B} = L_k(B \otimes B)D_k$ and $\bar{A} = L_k(A \otimes A)D_k$, C is $(k^* \times 1)$ vector, while B and A are $(k^* \times k^*)$ matrices. The elimination and duplication matrices, L_k and D_k are operational matrices of zeros and ones, so the parameters in (4) are just linear transformation from the equation (2).

2.3 Covariance targeting

The covariance targeting parameterization was introduced in Engle and Mezrich (1996) for the univariate GARCH model. The usefulness of covariance targeting in large models is that it allows for 2-step estimation which estimate the unconditional moments with empirical moments to reduce the number

of parameters to be estimated. Since unconditional covariance of daily return and realized return are $Var[r_t] = E[H_t] = P$ and $E[X_t] = M$. In 2, the covariance-targeting for H_t :

$$H_t = (P - BPB' - AMA') + B H_{t-1} B + A X_{t-1} A \quad (5)$$

assuming $(P - BPB' - AMA')$ is positive semidefinite. From 5, it is difficult to specify sufficient parameter restrictions to ensure that $(P - BPB' - AMA')$ is positive semidefinite. This feature has restrained us from estimate more flexible dynamics in this model other than just assume A and B are scalars. We could fit the model with rotated variables to extend the idea of covariance targeting in multivariate models of any dimension. Specifically, since $r_t = P^{1/2} z_t$, we define the rotated return as

$$\begin{aligned} z_t &= P^{-1/2} r_t \\ Var(z_t) &= I_k \end{aligned} \quad (6)$$

The conditional covariance of the rotated returns is $Var[z_t | \mathcal{F}_{t-1}] \equiv \tilde{H}_t$, noting that $H_t = P^{1/2} \tilde{H}_t P^{1/2}$. Similarly, for realized measure of covariance, we have $X_t = M^{1/2} \tilde{X}_t M^{1/2}$. So, we got $\tilde{H}_t = P^{-1/2} H_t P^{-1/2}$, $\tilde{X}_t = M^{-1/2} X_t M^{-1/2}$. Basically, we could scale the Realized MGARCH model with the unconditional covariance matrices. A covariance-targeting realized MGARCH model is:

$$\begin{aligned} \tilde{H}_t &= (I_k - BB' - AA') + B \tilde{H}_{t-1} B + A \tilde{X}_{t-1} A \\ E[\tilde{H}_t] &= E[\tilde{X}_t] = I_k \end{aligned} \quad (7)$$

This model has 2k dynamic parameters when B and A are diagonal. It is easy to impose the positive semidefinite of $(I_k - BB' - AA')$ for this specification. For the conditional covariance matrix of unrotated returns H_t , the BEKK model in 6 and 7 implies that

$$H_t = P^{1/2} \tilde{H}_t P^{1/2} = \tilde{C} \tilde{C}' + \tilde{B} H_{t-1} \tilde{B}' + \tilde{A} X_{t-1} \tilde{A}',$$

where

$$\tilde{B} = P^{1/2} B P^{-1/2}, \quad \tilde{A} = P^{1/2} A M^{-1/2}, \quad \tilde{C} \tilde{C}' = P^{1/2} (I_k - BB' - AA') P^{1/2} \quad (8)$$

It is clear the rotated BEKK implies a BEKK specification for the original variables in a constrained version since \tilde{A} and \tilde{B} depend on the spectral decomposition of P and M. Bauwens et.al(2006) discuss the invariance of multivariate models to linear transformation of return. In their definition, invariance means the transformed model is in the same series and keep the same dynamic specification(e.g scalar,diagonal

or full parameterization). In the case of diagonal specification, our model is clearly not invariant to linear transformation. Suppose A is diagonal, then from 8 \tilde{A} is a full asymmetric matrix, which means we get a full parameterized GARCH model for the unrotated variables. For the rotated model given in 6 and 7, we could easily impose the condition to ensure the long-run covariance stationary and also the positive definite of $(I_k - BB' - AA')$.

Assumption 1 *In the Realized rotated MGARCH model given by (6) and (7), $\rho(\tilde{A} \otimes \tilde{A} + \tilde{B} \otimes \tilde{B}) \leq 1$.*

This estimation is easily to impose in the estimation.

2.4 Calculate the realized kernel

3 Estimation and Inference

3.1 Estimation for MGARCH Equation

In this section we discuss the asymptotic properties of the quasi-maximum likelihood estimator within the realized MGARCH model given by (1) and (2). We could estimate the parameters in rotated model, then reverse the linear transformation to retract the unrotated measure. In traditional GARCH MODELS, the vector of daily returns is usually modeled as $R_t = Ht^{1/2}z_t$, with $z_t \stackrel{i.i.d}{\sim} \mathcal{N}(0, I_k)$, which motivates quasi-maximum likelihood estimation (QMLE). In our model, since $\xi_t = z_t z_t'$, the assumption that $z_t \stackrel{i.i.d}{\sim} \mathcal{N}(0, I_k)$ implies that ξ follows a Wishart distribution. The choice of Wishart distribution is suitable for the model where the support of the random variable of interest is confined to the space of positive semidefinite matrices.

The Realized MGARCH equation is parameterized with a finite-dimensional $(\delta \times 1)$ parameter vector $\theta \in \Theta \subset \mathbb{R}^\delta$. Decompose $\theta = (\theta'_s, \theta'_d)'$, where $\theta_s = (p', m')'$ denote the model's unconditional moment, and let θ_d denote the vector of dynamic parameters indexing \tilde{A} and \tilde{B} in (8). We denote the true parameter vector with θ_0 . The parameters are estimated in 2 steps. The static parameters are to be estimated by a moment estimator

$$p' = \text{vech}(P), \quad m' = \text{vech}(M)$$

where

$$\hat{P} = T^{-1} \sum_{t=1}^T r_t r_t', \quad \hat{M} = T^{-1} \sum_{t=1}^T X_t$$

This estimate is then decomposed into $P^{1/2}$ and $M^{1/2}$. Then we construct the time series of rotated returns and realized kernel $\hat{z}_t = \hat{P}^{-1/2}r_t$, $\hat{X}_t = \hat{M}^{-1/2}X_t\hat{M}^{-1/2}$, $t = 1, 2, \dots, T$. Then θ_d will be estimated by QMLE in the second step. The asymptotics of the QML estimator in this 2-step procedure is a direct application of two-step GMM estimation discussed in Newey and McFadden (1994).

The log-likelihood for the t^{th} observation will be denoted by $l_t(\theta)$. Inference for the Realized MGARCH equation will be based on QMLE of the following log-likelihood function of Wishart density which is similar with HEAVY model in Noureldin, etc (2011).

$$l_t(\theta_d; \theta_s) = C - \frac{1}{2}(\log |\tilde{H}_t| + \text{tr}(\tilde{H}_t^{-1} z_t z_t')) \quad (9)$$

where C is a constant with respect to θ . We assume the initial values H_0 is known and is positive semidefinite. The QML estimator is $\hat{\theta}_d$, where

$$\hat{\theta}_d = \arg \max_{\theta_d \in \Theta} L_t(\theta_d; \theta_s) = \arg \max_{\theta_d \in \Theta} \sum_{t=1}^T l_t(\theta_d; \theta_s)$$

To compute the asymptotic standard errors, we construct a group of conditions of moment for each t^{th} observation, $g_t = (g_{p,t}, g_{m,t}, g_{\theta_d,t})$, where $g_{p,t} = p - \text{vech}(r_t r_t')$, $g_{m,t} = p - \text{vech}(X_t)$ and $g_{\theta_d,t} = \frac{\partial \log L_t(\theta_s, \theta_d)}{\partial \theta_d}$. In the true parameter values, $E[g_t | \mathcal{F}_{t-1}] = 0$. Under standard regularity conditions, as $T \rightarrow \infty$ we have

$$\sqrt{T}(\hat{\theta} - \theta_0) \xrightarrow{d} N(0, \mathcal{I}^{-1} \mathcal{J} \mathcal{I}^{-1}) \quad (10)$$

where $\hat{\theta} = (\hat{\theta}_s', \hat{\theta}_d')$, and

$$\mathcal{J} = \text{Var} \left[\frac{1}{\sqrt{T}} \sum_{t=1}^T g_t \right], \mathcal{I} = E \left[\frac{\partial g_t}{\partial \theta} \right] \quad (11)$$

4 Adaptive LASSO

In this section, we would show that the adaptive lasso can be used to estimate the measurement equation in the Realized MGARCH model.

4.1 Definition and Notation

Since the measurement equation (3) is a fully parameterized linear model, the coefficient matrix has k^{*2} parameters to estimate, where $k^* = k(k+1)/2$. In fact, the equation (3) could be decomposed to a

group of equations seemingly unrelated, each with one dependent variable corresponding to elements of realized matrix X_t which is a symmetric matrix. So we got k^* univariate linear equations. Each equation has k^* parameters to links the realized covariance with all the conditional covariance cross different assets. In practice, the coefficient Γ is usually a sparse matrix around the identity matrix. We could use adaptive Least Absolute Selection and Shrinkage Operator (lasso) to select most significant variable and reduce others to zero to achieve the parsimony of the model.

The lasso is a regularization technique for simultaneous estimation and variable selection in linear regression (Tibshirani 1996). It just means estimating the Least Squares parameters subject to an L1 norm penalty to the slackness parameter. Since Meinshausen and Bhlmann (2004) also showed the conflict of optimal prediction and consistent variable selection in the lasso. Zou(2006) also has shown that lasso estimator is possibly inconsistent, we will adopt the adaptive lasso which enjoys the oracle properties by utilizing the adaptively weighted L1 penalty proposed in the same paper.

We define adaptive lasso(lasso) estimates β^{alasso} are given by

$$\beta^{alasso} = \arg \min_{\beta} \left\| y - \sum_{j=1}^p x_j \beta_j \right\|^2 + \lambda_n \sum_{j=1}^p \hat{w}_j |\beta_j| \quad (12)$$

\hat{w}_j is a vector of weights, with $\hat{w}_j = 1/\|\hat{\beta}\|$, where β is the unpenalized OLS estimator. Note equation (12) is a convex optimization problem, and thus it does not suffer from the multiple local minimal issue, and its global minimizer can be easily found after the alasso solution path being efficiently computed by the LARS algorithm (Efron et al., 2004), which only require computational effort of one single OLS fit.

Based on a simpler LARS algorithm proposed by Rosset and Zhu(2007), the computation details of alasso estimator are presented below:

4.2 Selective Criterion

In the alasso algorithm, we have to choose a criterion to reduce the number of variables. The question of how to find a criterion involves the trade-off between bias and variance, along with the subjective desire for parsimony.

The intention of using measurement equation is to update realized volatility measure beyond one step to make multiple-period prediction. Thus the accuracy of prediction is our main concern. We can not use RSS on the training data to determine these parameters, since we would always pick those that

Algorithm 1: Adaptive lasso algorithm

- 1 create new covariates $X_j^* = X_j/\hat{w}_j, j = 1, 2, \dots, q$. where \hat{w}_j is the adaptive weight as defined above. Note both X_j, Z_j^* are of dimension $n \times 1$, for each j .
 - 2 Solve lasso problem defined above in (12) via LARS algorithm for given λ , which corresponding to a step in the algorithm, each step add one variable.
 - 3 Then we substitute lasso estimator $\tilde{\beta}(\lambda)$ from step 2 in criterion equation below. This provides us a BIC value for each step.
 - 4 Repeat steps 2-3 for each remaining and record each BIC_λ for given $\tilde{\beta}(\lambda)$
 - 5 Choose the pair of $\tilde{\beta}(\lambda)$, which minimized BIC over λ
 - 6 Output is $\hat{\beta}_j = \tilde{\beta}_j/\hat{w}_j, j = 1, 2, \dots, q$. This is the adaptive lasso estimate.
-

fit most parameters and hence gave smallest residuals. Such models are unlikely to predict future data well due to larger variance. The fully parameterized OLS estimate is derived from taking least squares, hence its RSS is smallest(see table1).

OLS	Shrinkage Based AIC	Shrinkage Based BIC
348.53	357.39	365.62

Table 1: Residual of Sum-of-Squares(RSS) of different estimate

We proceed with 2 popular model selection criteria-Akaike information criterion (AIC) and Bayesian information criterion (BIC) for LASSO proposed by Zou(2006).

$$AIC^{(p)} = \left\| y - X\beta^{(p)} \right\|^2 + 2\sigma_\epsilon^2 df^{(p)} \quad (13)$$

$$BIC^{(p)} = \left\| y - X\beta^{(p)} \right\|^2 + \log(n)\sigma_\epsilon^2 df^{(p)} \quad (14)$$

Which Efron et al. (2004) showed that the number of degrees of freedom at each step of the LAR algorithm is well approximated by the number of non-zero elements of parameters β , $df_{LAR}^p = p, p = 1, \dots, k^*$. The measure σ_ϵ^2 represents the residual variance of OLS model, $\sigma_\epsilon^2 = \frac{1}{n-k} \|Y - X'\beta^{ols}\|^2$.

These two criterion has different asymptotic optimality. It is well known that AIC tends to select the model with the optimal prediction performance, while BIC tends to identify the true sparse model if the true model is in the candidate list (Hastie, and Tibshirani 2007). The L1 norm restriction parameter controls the penalty level and therefore the model complexity. Since LARS algorithm solves the LASSO

solution for all λ . With the entire LASSO solution path computed by LARS algorithm, we can immediately select the optimal model by identify the smallest value of selected criterion and the corresponding λ . We compare the two different criterion and their relationship of relative bound with df and RSS of model below.

In LASSO algorithm, an alternate tuning parameter t is used. t is called lasso parameter, which is also recognized as the absolute bound of estimated parameters $t = \sum \hat{\beta}_i^{alasso}$. It posses the one-to-one relationship with λ . Here we define standardized version parameter, s , as the relative bound $s = \sum \hat{\beta}_i^{alasso} / \sum \hat{\beta}_i^{ols}$, $s \in [0, 1]$. The relative bound can be seen as a standardized version of lasso parameter.

4.3 Covariance of adaptive lasso

The formula for variance of ALASSO is very complicated as showed in Zou(2006).

$$cov(\hat{\beta}_{\mathcal{A}}) = \sigma^2 (X'_{\mathcal{A}} X_{\mathcal{A}} + \lambda_n \sum (\hat{\beta}_{\mathcal{A}}))^{-1} X'_{\mathcal{A}} X_{\mathcal{A}} (X'_{\mathcal{A}} X_{\mathcal{A}} + \lambda_n \sum (\hat{\beta}_{\mathcal{A}}))^{-1}$$

where \mathcal{A} is the active set including all variable selected by alasso, and $\sum(\beta) = diag(\frac{\hat{u}_1}{|\beta_1|}, \dots, \frac{\hat{u}_k}{|\beta_k|})$. σ^2 is the variance of redidual of OLS fit which we could easily estimate with sample variance. The difficult part is that λ is hard to extract from LARS algorithm. According to the fundamental theorem of statistics, The empirical distribution (ED) of a set of independent drawings of a random variables generated by some data generating process (DGP) converges to the true CDF of the under the DGP. Using this theorem, we could calculate the sample covariance with bootstrapping method proposed by Efron(1979). Bootstrapping is the practice of estimating the properties of an estimator -say, its variance- by measuring those properties when sampling from an approximating distribution (the bootstrap DGP). Since we dont know the true DGP generated this parameters, we could approximate it with ED. The algorithm is shown below

5 Empirical illustration

The model is applied to high-frequency prices for ten stocks from the financial sector of the S&P 100

¹.These are: xxxxx. The sample period is mm/dd/yyyy to mm/dd/yyyy and the source of data is the TAQ database. We focus on the noise-robust realized kernel of Barndorff-Nielsen et al.(2011) as our choice

¹The ticker symbols are xxxxx

Algorithm 2: The Bootstrap algorithm

- 1 Select N pairs of independent bootstrap samples $(y_1^*, x_1^*), (y_2^*, x_2^*), \dots, (y_n^*, x_n^*)$ with replacement from original data
 - 2 Estimate the alasso parameters for each bootstrap sample $\hat{\tau}(n), n = 1, 2, \dots, N$.
 - 3 Estimate the standard error $se(\hat{\tau})$ by the empirical moment of the N group of parameters
 - 4 $s\hat{e}_N = \left[\frac{1}{N-1} \sum_{n=1}^N \{\hat{\tau}(n) - \bar{\hat{\tau}}\} \right]$ with $\bar{\hat{\tau}} = \frac{1}{N} \sum_{n=1}^N \hat{\tau}(n)$
-

for X_t . For the daily return and realized kernel, we focus on open-to-close returns which ignore the overnight effect.

5.1 Data Description

5.2 Realized Volatility Description

5.3 Daily Log-Return Description

5.4 Estimation for realized GARCH model

5.5 Adaptive lasso under different selective criterion

We index the lasso estimate with s to illustrate the shrinkage process in two figures. Figure (1) shows the LASSO estimates as a function of standardized bounds. LASSO continuously shrinks the coefficients toward zero as s decreases; and some coefficients shrink to exactly zero if the constraints is sufficiently large. In figure (2), the number of variables decreases as relative bound decreases (penalty constraints increases). In both figures, We show how different the model will be by selecting variables according to different criterion. The broken line represents AIC criterion, we select model at $AIC=362, s=0.32$; The solid line represents BIC criterion, we select model at $BIC=371, s=0.05$. The value of s validates the sparsity of true model. Roughly, this corresponds to keeping just 30% of the predictors. In the figure (3), the mean square of error of alasso estimate decreases as relative bound decreases (penalty constraints increases). We could find an elbow shape in the curve, which means part of parameters don't contribute a lot for reducing the MSE. As a result, the optimal selection value of s always happens near the elbow area.

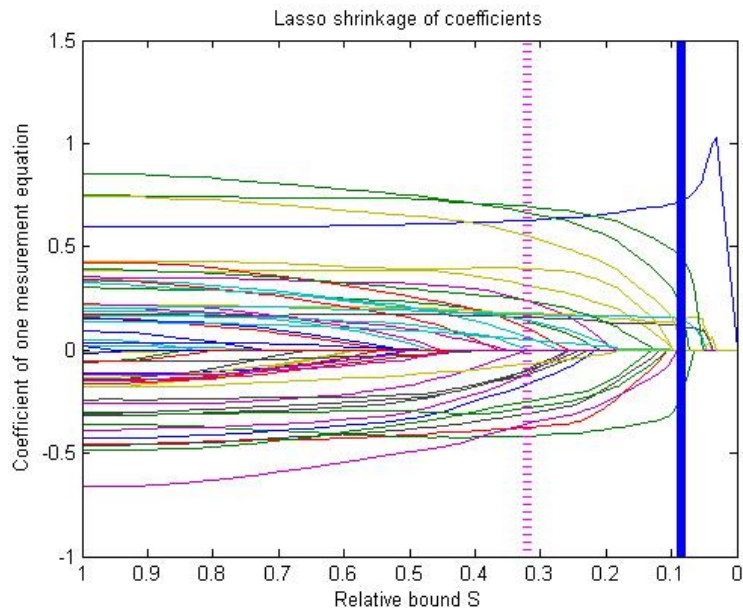


Figure 1: Lasso shrinkage of coefficient

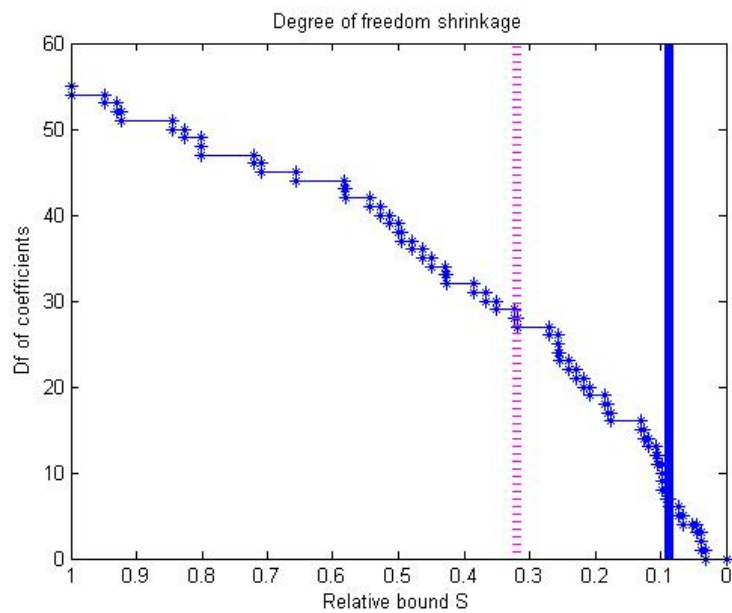


Figure 2: Degree of freedom shrinkage

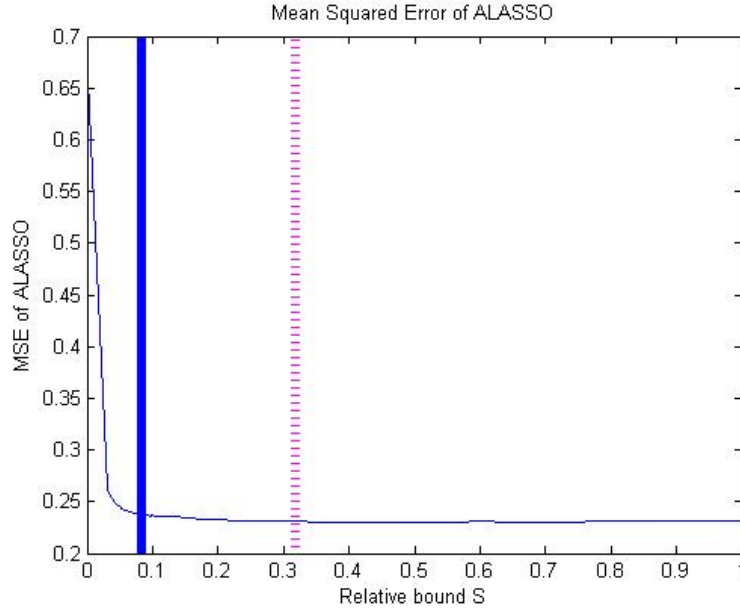


Figure 3: Mean Squared Error of ALASSO

5.6 Estimation of Measurement equation

With the structure of measurement equations, each realized variance (covariance) is the function of corresponding conditional variance (covariance) and the conditional covariance of pairs of all other assets. Assume there are three asset A, B and C, the realized variance of asset A is updated by the conditional variance of A and conditional covariance of A and B, the one of A and C, and also the one of B and C. Apparently the conditional variance of asset A should be the main driving power in updating the realized variance of asset A. So the coefficient of conditional variance of asset A should be larger than other ones. As a result, the coefficient matrix should have larger elements in the diagonal position. Most elements that alasso shrinks are in the off-diagonal position. Even some diagonal elements will also shrink since the lagged driving force in updating covariance is smaller than the one of variance. In the case of 10 assets, we got coefficient matrix in size 55*55, in the diagonal position the elements in 1, 11, 20, 28, 35, 41, 46, 50, 53, 55 rows are near much larger than others. In the appendix, we compared first row (other rows are similar) and diagonal elements of coefficient matrix by using OLS and alasso using AIC and BIC criterion, the number in parenthesis are standard errors.

6 Conclusion

Appendix

Table 2: Parameter Estimates of the "A" matrix in rotated realized MGARCH model

	ABT	DOW	IBM	AXP	BK	C	CAT	COF	USB	WFC
ABT	0.7213 (0.0627)									
DOW		0.4775 (0.0150)								
IBM			0.5150 (0.0503)							
AXP				0.6125 (0.0308)						
BK					0.5311 (0.0143)					
C						0.6585 (0.0011)				
CAT							0.4642 (0.0361)			
COF								0.6701 (0.0154)		
USB									0.7723 (0.0292)	
WFC										0.5135 (0.0089)

Table 4: Estimation of Γ

Position in Coefficient Matrix	First row			Diagonal		
Model selection criterion	LS	AIC	BIC	LS	AIC	BIC

Continued on next page

Table 4: Estimation of Γ [illegible]

Continued on next page

Table 4: Estimation of Γ [illegible]

Continued on next page

Table 3: Parameter Estimates of the "B" matrix in rotated realized MGARCH model

	ABT	DOW	IBM	AXP	BK	C	CAT	COF	USB	WFC
ABT	0.5835 (0.1073)									
DOW		0.8747 (0.0049)								
IBM			0.8012 (0.0402)							
AXP				0.7454 (0.0271)						
BK					0.8321 (0.0068)					
C						0.7512 (0.0003)				
CAT							0.8547 (0.0199)			
COF								0.7172 (0.0141)		
USB									0.6006 (0.0453)	
WFC										0.8514 (0.0032)