Scalable inference for a full multivariate stochastic volatility model

Petros Dellaportas[*], Michalis K. Titsias[†], Katerina Petrova[‡], and Anastasios Plataniotis[§]

Abstract

We introduce a multivariate stochastic volatility model that imposes no restrictions on the structure of the volatility matrix and treats all its elements as functions of latent stochastic processes. Inference is achieved via a carefully designed feasible and scalable MCMC that has quadratic, rather than cubic, computational complexity for evaluating the multivariate normal densities required. We illustrate how our model can be applied on macroeconomic applications through a stochastic volatility VAR model, comparing it to competing approaches in the literature. We also demonstrate how our approach can be applied to a large dataset containing 571 stock daily returns of Euro STOXX index.

Key Words: Bayesian Analysis; Computational complexity; Givens angles; MCMC; Time-varying parameter vector autoregressive.

JEL Classification: C11,C13

[*]Corresponding author. Department of Statistical Science, University College, Gower Street, London WC1E 6BT, UK and Department of Statistics, Athens University of Economics and Business, Patission 76, Athens 10434, Greece. email: p.dellaportas@ucl.ac.uk

[†]Department of Informatics, Athens University of Economics and Business, Greece

[‡]Barcelona School of Economics, Pompeu Fabra University, Barcelona, Spain

[§]Ernst and Young, Athens, Greece

1 Introduction

We aim to model a sequence of high dimensional $N \times N$ volatility matrices $\{(\Sigma_t)_{t=1}^T\}$ of an N-dimensional zero mean, normally distributed, time series vector of asset returns $\{(r_t)_{t=1}^T\}$. The prediction of Σ_{T+1} is a fundamental problem in financial statistics that has received a lot of attention in portfolio selection and financial management literature, see for example Tsay (2005). The major statistical challenge emanates from the fact that each Σ_t is positive-definite and its number of parameters grows quadratically in N. A popular paradigm in financial econometrics is to adopt observational-driven models that extend the popular univariate GARCH-type formulations, see for example Engle (2002). In these models parameters are deterministic functions of lagged dependent variables so are perfectly predictable one-step-ahead given past information. We focus, instead, on parameter driven models that assume that elements of $\{\Sigma_t\}$ vary over time as dynamic processes with idiosyncratic innovations.

The starting point in our model construction is the one-dimensional stochastic volatility model introduced by Taylor (1986) which allows the log-volatility of the observations to be an autoregressive unobserved random process. The challenging extension to the multivariate case is discussed in the reviews by Platanioti et al. (2005), Asai et al. (2006) and Chib et al. (2009). Due to both the computational complexity that increases dramatically with N and the modelling complexity produced by the necessity to stochastically evolve correlations and volatilities preserving the positive definiteness of Σ_t, all existing models assume some form of model parsimony that often correspond to the simplifications suggested in the observation driven models literature. In particular, the existing multivariate stochastic volatility (MSV) models assume either constant correlations over time or some form of dynamic correlation modelling through factor models with factors being independent univariate stochastic volatility models; see, for example, Harvey et al. (1994), Kim et al. (1998a), Pitt and Shephard (1999a), Bauwens et al. (2006), Tims and Mahieu (2003). Different approaches to MSV models have been suggested by Philipov and Glickman (2006a,b) who propose modelling Σ_t as an inverted Wishart process, by Gouriéroux et al. (2009) who introduced the Wishart autoregressive process, and by Carvalho et al. (2007) who suggest dynamic matrix-variate graphical models.

We propose a new MSV modelling formulation which is full in the sense that all $N(N+1)/2$ elements

of Σ_t evolve in time. A key idea of our approach is to assume Gaussian latent processes for functions of the eigenvalues and rotation angles of Σ_t. By invert-transforming back to Σ_t the positive definiteness is immediately ensured. For a N-dimensional vector of responses, we construct a MSV model with $N(N+1)/2$ Gaussian latent paths corresponding to N eigenvalues and $N(N-1)/2$ rotation angles. When N is prohibitively large, we also propose a dynamic factor extension of our model in which the volatility matrices of the factors are treated as the stochastic dynamically evolving $\{\Sigma_t\}_{t=1}^T$ in our baseline MSV model. This generalises the existing assumption of factor independence that is prominent in dynamic factor models in many statistical areas including, in addition to financial econometrics, also economics, see for example Forni et al. (2000), and psychology, see for example Ram et al. (2013).

While our model formulation allows for the construction of latent processes ensuring the positive definiteness of $\{\Sigma_t\}$ in theory, the estimation process remains a computationally challenging task. In practical quantitative finance areas, such as portfolio construction and risk management, interest lies in applications where the number of assets N is in the size of hundreds. Our approach is Bayesian so our view of the problem is that we deal with a non-linear likelihood function with a latent $TN(N+1)/2$-dimensional Gaussian prior distribution. Since the likelihood itself requires evaluation of a $TN(N+1)/2$-dimensional Gaussian density, computational efficiency is a major impediment not only because of the cubic computational complexity required to perform the Gaussian density matrix manipulations, but also because Markov chain Monte Carlo (MCMC) algorithms require carefully chosen simultaneous updates of the latent paths so that good chain mixing is achieved.

An important contribution of the paper is that our proposed Bayesian inference is carefully designed to handle both these problems. The crucial MCMC moves that update the latent paths are based on an auxiliary gradient-based sampler suggested by Titsias (2011) and further developed in Titsias and Papaspiliopoulos (2018). Moreover, we provide algorithms that achieve computational complexity of squared, rather than cubic, order for the evaluation of the likelihood Gaussian density and its derivative with respect to rotation angles and eigenvalues. This overcomes a crucial impediment that is common in many multivariate statistics applications, see for example Banerjee et al. (2008) for a recent review of this problem in spatial statistics.

We extend our methodology to modelling stochastic volatility in vector autoregressive (VAR) models which are commonly used in applied macroeconomics. The main advantage of our approach is that it permits estimation of the time-varying covariance matrix of the reduced-form VAR model that is invariant to the ordering of the variables in the system unlike the prevailing time-varying Cholesky specification. We show that the resulting estimates of our model can differ considerably from the Cholesky specification. Moreover, we illustrate in a simple three-variable system how different permutations of the variables can lead to different empirical conclusions when the Cholesky time-varying specification is used, which is undesirable particularly when the model is used for reduced-form analysis, such as forecasting.

Finally, we apply our method to a computationally challenging, financial big data example based on ten years daily returns of 571 stocks of the Euro STOXX index. We formulate a factor MSV model and evaluate the predictive ability of a series of models by gradually increasing the number of factors and evaluating the distance between the predictive volatility matrix and the quadratic co-variation of the next day based on 5-minutes intra-day data.

The rest of the paper is organised as follows. Section 2 presents the baseline model and Section 3 incorporates it to a full factor MSV model. Section 4 describes the parameter estimation procedure, in Section 5 we discuss the computational complexity of our proposed algorithm and compare it to competing approaches, and Section 6.1 introduces a time-varying volatility VAR model that incorporates our MSV approach to modelling the covariance of the VAR residuals and applies it to macroeconomic time series. Section 6.2 illustrates the computational power of our algorithm by applying it to a 571 asset returns and illustrates its superior predictive ability over competing models. We conclude with a small discussion in Section 7.

2 The baseline multivariate stochastic volatility model

We assume that an observed vector time series r_t satisfies $r_t|\Sigma_t \sim \mathcal{N}(0, \Sigma_t)$ and is covariance stationary so $E(\Sigma_t) = \Sigma$ exists and is a symmetric p.d. matrix. The spectral decomposition $\Sigma_t = P_t \Lambda_t P_t^T$ parameterises the $N(N+1)/2$ independent time-changing entries of Σ_t to N eigenvalues $\{(\Lambda_{it})_{i=1}^N\}$ and $N(N-1)/2$

parameters in the eigenvector matrices P_t. We further write each P_t as a product of $N(N-1)/2$ Givens rotation matrices $P_t = \prod_{i<j} G_{ij}(\omega_{ij,t})$ where the elements of each Givens matrix $G_{ij}(\omega_{ij,t})$ are given by

$$G_{ij}[k,l] = \begin{cases} \cos(\omega_{ij,t}), & \text{if } k=l=i \text{ or } k=l=j \\ \sin(\omega_{ij,t}), & \text{if } k=i, l=j \\ -\sin(\omega_{ij,t}), & \text{if } k=j, l=i \\ 1, & \text{if } k=l \\ 0, & \text{otherwise.} \end{cases}$$

Each rotation matrix has one parameter, the rotation angle $\omega_{ij,t}$, which appears in only four cells of the matrix. For each time t there are $N(N-1)/2$ angles $\{(\omega_{ij,t})_{i<j}\}$ associated with all possible pairs (i,j) where $i<j, j=1,\ldots,N$. We choose $\omega_{ij,t} \in (-\pi/2, \pi/2)$ to ensure uniqueness of the rotation angles and we transform angles and eigenvalues to $\delta_{ij,t} = \log(\pi/2+\omega_{ij,t}) - \log(\pi/2-\omega_{ij,t})$ and $h_{i,t} = \log(\Lambda_{it})$. Our proposed MSV model is

$$\begin{aligned} h_{i,t+1} &= h_{i,0} + \phi_i^h \cdot (h_{i,t} - h_{i,0}) + \sigma_i^h \cdot \eta_{i,t}^h,\ i=1,\ldots,N,\ t=1,\ldots,T-1, \\ \delta_{ij,t+1} &= \delta_{ij,0} + \phi_{ij}^\delta \cdot (\delta_{ij,t} - \delta_{ij,0}) + \sigma_{ij}^\delta \cdot \eta_{ij,t}^\delta,\ i<j,\ t=1,\ldots,T-1, \\ h_{i,1} &\sim \mathcal{N}\left(h_{i,0}, \frac{(\sigma_i^h)^2}{1-(\phi_i^h)^2}\right), \quad \delta_{ij,1} \sim \mathcal{N}\left(\delta_{ij,0}, \frac{(\sigma_{ij}^\delta)^2}{1-(\phi_{ij}^\delta)^2}\right), \end{aligned} \tag{1}$$

where $|\phi_i^h| < 1$ and $|\phi_{ij}^\delta| < 1$ are the persistence parameters of each autoregressive process, σ_i^h and σ_{ij}^δ are corresponding error variances and $\eta_{i,t}^h, \eta_{ij,t}^\delta \sim \mathcal{N}(0,1)$ independently. Note that due to time-changing prior structure in (1) our prior is not orthogonally invariant. The parameter vectors that need to be estimated are the transformed rotation angles and eigenvalues $\{(\delta_t)_{t=1}^T\}$,$\{(h_t)_{t=1}^T\}$, and the latent path parameters $\theta_h = \{(\phi_i^h, h_{i,0}, \sigma_i^h)_{i=1}^N\}$ and $\theta_\delta = \{(\phi_{ij}^\delta, \delta_{ij,0}, \sigma_{ij}^\delta)_{i<j}\}$ related to transformed eigenvalues and rotation angles respectively. The volatility matrices Σ_t are positive definite since they are obtained by just transforming back the parameters h_t, δ_t to P_t and Λ_t. The parametric prior specification in (1) is a modelling choice and our approach allows for different prior specifications, dependeing on the application at hand.

Givens angles have been used in the past in Bayesian literature in static problems with focus on improvement of covariance matrix estimation via reference or shrinkage priors; see Yang and Berger (1994) and Daniels and Kass (1999). The effect of left-multiplying a Givens matrix $G_{ij}(\omega_{ij,t})$ to a vector is to rotate the vector clockwise by $\omega_{ij,t}$ radians in the plane spanned by the ith and jth components of the vector. The covariance matrix Σ_t can therefore be viewed as that of a vector of N uncorrelated random variables with variances $(\Lambda_{it})_{i=1}^{N}$, rotated successively by applying Givens rotation matrices. Sparsity may be induced by setting many angles equal to zero since when a rotation angle is zero there is no rotation in the corresponding plane. Cron and West (2016) exploit this fact and proposed sparsity modelling of a covariance matrix by placing priors on the Givens angles.

When the assumption of exchangeability between the asset returns is plausible, we suggest using a hierarchical formulation of the form

$$\begin{aligned} \phi_i^h &= (e^{\tilde{\phi}_i^h} - 1)/(e^{\tilde{\phi}_i^h} + 1) \\ \tilde{\phi}_i^h | \mu_h, \lambda_h &\sim \mathcal{N}(\mu_h, \lambda_h^{-1}) \\ (\mu_h, \lambda_h) &\sim \mathcal{N}(\mu_0, (k_0\lambda_h)^{-1})\text{Ga}(\alpha_0, \beta_0). \end{aligned} \tag{2}$$

In the large-dimensional financial applications we are interested in, this prior specification has great practical importance. In all large portfolios there are assets with fewer observations due to new stock introductions to the market or to an index, mergers and acquisitions, etc. In these cases, the Bayesian hierarchical model allows borrowing strength from the persistence parameters which results to their shrinkage towards the overall mean μ_h. Of course, other assumptions such as exchangeability within markets or sectors might be more appropriate and the prior specification may be chosen accordingly. We propose non-informative prior densities for θ_h and θ_δ by placing an inverse Gamma density for $(\sigma_i^h)^2$ and $(\sigma_{ij}^\delta)^2$ and an uninformative uniform improper prior density for $h_{i,0}$ and $\delta_{ij,0}$. Further details, such as the values of the hyperparameters used in our simulations and real data, are given in the Supplementary material.

3 The full factor MSV model

The basic model (1) can be extended to a full factor MSV model by assuming that the means of the initial series r_t are linear combinations of K factors which are modelled as MSV processes. This can be written as $r_t = Bf_t + V^{1/2}\epsilon_t$, and $f_t|\Sigma_t \sim \mathcal{N}(0, \Sigma_t)$ where B is a $N \times K$ matrix of factor loadings, f_t is a K-dimensional vector that is modelled with the MSV model (1), V is an $N \times N$ diagonal matrix of variances v_i and ϵ_t is a vector of N independent $\mathcal{N}(0, 1)$ variates. For identification purposes, constraints on the elements b_{ij} of B must be imposed, so we set $b_{ij} = 0$ for $i < j, i \leq K$ and $b_{ii} = 1$ for $i \leq K$. The covariance of r_t at time t is separated into systematic and idiosyncratic components $B\Sigma_t B^T + V$. The non-zero values of the factor loadings matrix B are assigned a conjugate Gaussian prior density while the noise variances v_i are assumed to have a standard conjugate inverse Gamma prior; see the Supplementary material for further details.

Existing factor MSV models in the financial econometrics literature assume that f_t are independent univariate stochastic volatility processes, a very unrealistic assumption given the broad empirical evidence on observed priced factors. We refer to these models as independent factor models. Our full factor model extension provides a generalisation by assuming that both the factor variances and the correlations evolve stochastically and it reduces to model (1) when $N = K$, $B = I$ and all $v_i = 0$; and to an independent factor model by setting all rotation angles equal to zero. Our approach is general and can be used to model the stochastic volatility of the idiosyncratic component $V^{1/2}\epsilon_t$; however, in that case, the factors f_t should be modelled with constant volatility. This approach is possible, but it will diminish the benefits of the factor structure for reducing the dimension of the stochastic volatility matrix.

4 Estimation

To estimate the parameters of the model we follow a fully Bayesian procedure by applying an MCMC algorithm. We describe here the algorithmic steps for the full factor MSV model noting that the steps for the simple MSV model are obtained as a special case. Suppose a sample of observed time series vectors $r_t \in$

$\mathbb{R}^N$ is available for $t = 1, \ldots, T$, which we wish to model by using a factor MSV model having K latent factors. While in the real application considered in Section 6.2 we do allow for missing values, for notational simplicity here we assume that the vectors r_t have no missing values (the treatment of missing values under the full factor model is straightforward as explained in the previous Section). The joint probability distribution of all observations, latent variables and parameters is written in the form

$$\left(\prod_{t=1}^{T} \mathcal{N}(r_t | Bf_t, VI) \mathcal{N}(f_t | 0, \Sigma_t(x_t)) \right) p(X|\theta_h, \theta_\delta) p(\theta_h, \theta_\delta) p(B, V),$$

where $x_t = \{(h_{i,t})_{i=1}^p, (\delta_{ij,t})_{i<j}\}$ denotes the $K(K+1)/2$ vector of all transformed angles and log-eigenvalues that determine the volatility matrix at time t. The expression $\mathcal{N}(r_t|Bf_t, VI)$ represents the density function $\mathcal{N}(Bf_t, VI)$ evaluated at r_t. Finally, $X = (x_1, \ldots, x_T)$ denotes the full set of latent variables, represented as a row-wise unfolded vector of the $K(K+1)/2 \times T$ matrix in which each T-dimensional row vector stores the latent variables associated with a specific Gaussian autoregressive process. Thus, $p(X|\theta_h, \theta_\delta)$ can be a high-dimensional Gaussian distribution, having an inverse covariance matrix with $K(K+1)/2$ separate blocks associated with the independent latent Gaussian processes and where each T-dimensional block has a sparse tridiagonal form.

Performing MCMC on the above model is extremely challenging due the very large state space. For instance, for a typical real-world dataset, as the one we consider in our financial application in Section 6.2, the number of latent variables in X can be of order of millions, for example for $K = 50$ and $T = 2000$ the size of X is 2,55 millions. To deal with such dimensions, we develop a well-mixing computationally scalable MCMC procedure that uses an effective move that jointly samples (in a single step) all random variables in X.

4.1 The general structure of the MCMC algorithm

The random variables we need to infer can be naturally divided into three groups: i) the full factor model parameters and latent variables $(B, V, f_1, \ldots, f_T)$ that appear in the observation likelihoods, ii) the MSV

latent variables X that determine the volatility matrices and iii) the hyperparameters $(\theta_h, \theta_\delta)$ that influence the latent Gaussian prior distribution $p(X|\theta_h, \theta_\delta)$. We construct a Metropolis-within-Gibbs procedure that sequentially samples each of the above three groups of variables conditional on the others. Schematically, this is described as

$$
\begin{aligned}
B, V, f_1, \ldots, f_T \leftarrow p(B, V, (f_t)_{t=1}^T|\text{rest}) &\propto \left(\prod_{t=1}^T \mathcal{N}(r_t|Bf_t, VI)\mathcal{N}(f_t|0, \Sigma_t(x_t))\right) p(B, V), \\
X \leftarrow p(X|\text{rest}) &\propto \left(\prod_{t=1}^T \mathcal{N}(f_t|0, \Sigma_t(x_t))\right) p(X|\theta_h, \theta_\delta), \\
\theta_h, \theta_\delta \leftarrow p(\theta_h, \theta_\delta|\text{rest}) &\propto p(X|\theta_h, \theta_\delta)p(\theta_h, \theta_\delta).
\end{aligned}
$$

The first step of sampling the full factor model parameters is further split into three conditional Gibbs moves for updating the factor loadings matrix B, the diagonal covariance V and the latent factors $f_1, \ldots, f_T$. This involves simulating from standard conjugate conditional distributions, whose explicit forms are given in the Supplementary material. However, the conjugate Gibbs step for sampling the latent factors $f_1, \ldots, f_T$ is very expensive for our application, as it scales as $O(TK^3)$. Therefore, we replace this step with a more scalable Metropolis within Gibbs step that costs $O(TNK)$ as we detail in Section 5.2. The third step of sampling θ_h and θ_δ also involves standard procedures: Gibbs moves for the parameters $h_{i,0}$, $\delta_{ij,0}$, $(\sigma_i^h)^2$, $(\sigma_{ij}^\delta)^2$ and Metropolis-with-Gibbs for the transformed persistence parameters of the AR processes; full details are given in the Supplementary material. The most challenging step in the above MCMC algorithm is the second one where we need to simulate X. This requires simulating from a latent Gaussian variable model where the high-dimensional X follows a Gaussian prior distribution $p(X|\theta_h, \theta_\delta)$ and then generates the latent factors $F = (f_1, \ldots, f_T)$ through a non-Gaussian density $p(F|X) = \prod_{t=1}^T \mathcal{N}(f_t|0, \Sigma_t(x_t))$, where X appears non-linearly inside the volatility matrices. We can think of $p(F|X)$ as the likelihood function in this latent Gaussian variable model where F plays the role of the observed data. To sample X we implement an efficient algorithm proposed by Titsias (2011) that we describe in Section 4.2 in detail.

We emphasize that the usual ordering of eigenvalues is not needed during the sampling process since

each sampled value of x_t reconstructs invariantly a sample for Σ_t. Finally, from a practical perspective, the most interesting posterior summary of the MCMC algorithm is the predictive density of Σ_{T+1} which is constructed by transforming all the predictive densities of x_{T+1} produced exactly as described in the very first paper on Bayesian estimation for univariate stochastic volatility models by Jacquier et al. (1994b).

4.2 Auxiliary gradient-based sampling for latent Gaussian variables models

The algorithm in Titsias (2011); Titsias and Papaspiliopoulos (2018) is based on combining the Metropolis-Adjusted Langevin Algorithm (MALA) with auxiliary variables in order to efficiently deal with a latent Gaussian variable model. The use of auxiliary variables allows to construct an iterative Gibbs-like procedure which makes efficient use of gradient information of the intractable likelihood $p(F|X)$ and is invariant under the tractable Gaussian prior $p(X|\theta_h, \theta_\delta)$. For the remainder of this Section we shall simplify notation by dropping reference to the parameters θ_h and θ_δ which are kept fixed when sampling X, so that the Gaussian prior is written as $p(X) = \mathcal{N}(X|M, Q^{-1})$, where M is the mean vector and Q is the inverse covariance matrix. Suppose that we are at the n-th iteration of the MCMC and the current state of X is X_n. We introduce auxiliary variables U that are of the same dimension as X and are sampled from the following Gaussian density conditional on X_n:

$$p(U|X_n) = \mathcal{N}(U|X_n + \frac{\zeta}{2}\nabla \log p(F|X_n), \frac{\zeta}{2}I),$$

where $\nabla \log p(F|X_n)$ denotes the gradient of the log-likelihood evaluated at the current state X_n. U injects Gaussian noise into the current state X_n and shifts it by $(\zeta/2)\nabla \log p(F|X_n)$, where ζ is a step size parameter. Thus, X_n is moved towards the direction where the log-likelihood takes higher values and $p(U|X_n)$ corresponds to a hypothetical MALA proposal distribution associated with a target density that is solely proportional to the likelihood $p(F|X)$. A difference, however, is that in this distribution the step size or variance is $\zeta/2$, while in the regular MALA the variance is ζ. This is because U aims at playing the role of an intermediate step that feeds information into the construction of the proposal density for sampling X_{n+1}. The remaining variance $\zeta/2$ is added in a subsequent stage, when a proposal is specified in a way that in-

variance under the Gaussian prior density is achieved. More precisely, if the target were just proportional to the likelihood $p(F|X)$, then we could propose a candidate state Y given U from $Y \sim \mathcal{N}(Y|U, \zeta/2)$ and by marginalizing out the auxiliary variable U we would have recovered the standard MALA proposal distribution $\mathcal{N}(Y|X_n + (\zeta/2)\nabla \log p(F|X_n), \zeta)$. However, since our actual target is $p(F|X)p(X)$ and $p(X)$ is a tractable Gaussian term, we modify the proposal distribution by multiplying it with this Gaussian distribution so that the whole proposal becomes invariant under the prior. The proposed Y is sampled from the proposal density

$$q(Y|U) = \frac{1}{\mathcal{Z}(U)}\mathcal{N}(Y|U, \frac{\zeta}{2}I)p(Y) = \mathcal{N}(Y|(I + \frac{\zeta}{2}Q)^{-1}(U + \frac{\zeta}{2}QM), \frac{\zeta}{2}(I + \frac{\zeta}{2}Q)^{-1})$$

where $\mathcal{Z}(U) = \int \mathcal{N}(Y|U, \frac{\zeta}{2}I)p(Y)dY$. A proposed Y is accepted or rejected with Metropolis-Hastings acceptance probability $\min(1, r)$ where

$$\begin{aligned} r &= \frac{p(F|Y)p(U|Y)p(Y)}{p(F|X_n)p(U|X_n)p(X_n)}\frac{q(X_n|U)}{q(Y|U)} = \frac{p(F|Y)p(U|Y)p(Y)}{p(F|X_n)p(U|X_n)p(X_n)}\frac{\mathcal{Z}(U)^{-1}\mathcal{N}(X_n|U, (\zeta/2)I)p(X_n)}{\mathcal{Z}(U)^{-1}\mathcal{N}(Y|U, (\zeta/2)I)p(Y)} \\ &= \frac{p(F|Y)\mathcal{N}(U|Y + (\zeta/2)D_y, (\zeta/2)I)}{p(F|X_n)\mathcal{N}(U|X_n + (\zeta/2)D_t, (\zeta/2)I)}\frac{\mathcal{N}(X_n|U, \frac{\zeta}{2}I)}{\mathcal{N}(Y|U, \frac{\zeta}{2}I)} \\ &= \frac{p(F|Y)}{p(F|X_n)}\exp\left\{-(U - X_n)^T D_t + (U - Y)^T D_y - \frac{\zeta}{4}(||D_y||^2 - ||D_t||^2)\right\} \end{aligned} \tag{3}$$

where $D_t = \nabla \log p(F|X_n)$, $D_y = \nabla \log p(F|Y)$ and $||.||$ denotes the Euclidean norm of a vector. An important observation in the resulting form of (3) is that the Gaussian prior terms $p(X_n)$ and $p(Y)$ have been cancelled out from the acceptance probability, so their evaluation is not required: the resulting $Q(Y|U)$ is invariant under the Gaussian prior. The basic sampling steps are summarised in Algorithm 1. A simplified

(i) $U \sim \mathcal{N}(U|X_n + (\zeta/2)D_t, (\zeta/2)I)$
(ii) $Y \sim \mathcal{N}(Y|(I + (\zeta/2)Q)^{-1}(U + (\zeta/2)QM), (\zeta/2)(I + (\zeta/2)Q)^{-1})$ and with probability $\min(1, r)$, where r is given by (3), $X_{n+1} = Y$ or otherwise $X_{n+1} = X_n$.

Algorithm 1: Auxiliary gradient-based sampler

version is obtained when we ignore the gradient from the likelihood $p(F|X)$. Then, the algorithm reduces to an auxiliary random walk Metropolis which is implemented exactly as Algorithm 1 with the only difference that the gradient vectors D_t and D_y are now equal to zero, leading to simplifications of some expressions; for example, the probability r reduces to the likelihood ratio. An elegant property of the above auxiliary sampling procedure is that when the Gaussian prior tends to a uniform distribution by letting $Q \to 0$, it proposes as standard MALA or as standard random walk Metropolis algorithms. This can be seen by observing that the marginal proposal distribution in step (ii) of Algorithm 1 reduces to the previous standard schemes where the underlying target distribution is proportional to the likelihood $p(F|X)$. This suggests that in order to set the step size parameter ζ we can follow the standard practice in adaptive MCMC, so that for the auxiliary gradient-based sampler we can tune ζ to achieve an acceptance rate of around $50-60\%$ and for the auxiliary random walk Metropolis an acceptance rate of $20-30\%$. Empirically, it has been found that these regions are associated with optimal performance, see Titsias and Papaspiliopoulos (2018).

In order to apply the above algorithm to the full factor MSV model where the size of X can be of order of millions, we have to make sure that the computational complexity remains linear with respect to the size of X. This is made possible because the Gaussian prior $\mathcal{N}(X|M, Q^{-1})$ has a sparse tridiagonal inverse covariance matrix Q. Thus, given that Q is tridiagonal, the matrix $(2/\zeta)I + Q$ will also be tridiagonal and similarly the matrix L obtained from the Cholesky decomposition $LL^T = (2/\zeta)I + Q$, will be a lower two-diagonal matrix which can be computed efficiently in linear time. Then, a sample Y in the step 2 of Algorithm 1 can be simulated according to

$$Y = L^{-T}(L^{-1}(\frac{2}{\zeta}U + QM) + Z), \;\; Z \sim \mathcal{N}(0, I),$$

where parentheses indicate the order in which the computations should be performed. All these computations, including the two linear systems needed to be solved, can be performed efficiently in linear time since the associated matrices are either tridiagonal or lower two-diagonal. Therefore, the overall complexity when sampling Y is linear with respect to the size of this vector. Since this vector has size $K(K+1)/2 \times T$ the computational complexity scales as $O(TK^2)$.

Finally, the above algorithm requires the evaluation of the acceptance probability which is dominated by the likelihood ratio that involves the density $p(F|X)$ given by (4.1) which consists of a product of T K-dimensional multivariate Gaussian densities. Furthermore, we need to compute gradients of the form $\nabla \log p(F|X)$ of this log likelihood that appear in the acceptance probability and are required also when sampling U. A usual computation of these quantities scales as $O(TK^3)$ which is too expensive for the real applications of the full factor MSV model. By taking advantage of the analytic properties of the Givens matrices we can reduce the computational complexity to $O(TK^2)$, that is quadratic with respect to dimensionality of the Gaussians. To achieve such a complexity, we have developed the specialized algorithms detailed in the next Section.

5 Computational complexity

5.1 $O(N^2)$ computation for the MSV model

A crucial property of the MSV model is that the evaluation of its log density and the corresponding gradients with respect to the parameters inside the volatility matrix Σ_t can be computed in $O(N^2)$ time. This differs from other more commonly-used parametrisations of the multivariate Gaussian distribution where computations scale as $O(N^3)$ and they are infeasible for large N. Assume that we wish to evaluate the log density associated with the vector $r_t \sim \mathcal{N}(0, \Sigma_t)$ written as

$$\log \mathcal{N}(r_t|0, \Sigma_t) = -\frac{N}{2}\log(2\pi) - \frac{1}{2}\sum_{i=1}^{N} h_{it} - \frac{1}{2} v_t^T v_t, \tag{4}$$

where $v_t = \Lambda_t^{-\frac{1}{2}} P_t^T r_t$ and where we used that $\log|\Sigma_t| = \log|\Lambda_t| = \sum_{i=1}^{N} h_{it}$. Clearly, given v_t the above expression takes $O(N)$ time to compute. Therefore, in order to prove $O(N^2)$ complexity we need to show that the computation of v_t scales as $O(N^2)$. This is based on the fact that the transformed vector $G_{ij}(\omega_{ji,t})^T v_t$ takes $O(1)$ time to compute since all of its elements are equal to the corresponding ones from the vector v_t apart from the i-th and j-th elements that become $v_t[i]\cos(\omega_{ji,t}) - v_t[i]\sin(\omega_{ji,t})$ and

$v_t[j]\sin(\omega_{ji,t})+v_t[j]\cos(\omega_{ji,t})$, respectively. Thus, the whole product with all $N(N-1)/2$ Givens matrices can be carried out recursively in $O(N^2)$ time as shown in Algorithm 2.

Initialize $v_t = r_t$.
For $i = 1$ to $N-1$
  For $j = i+1$ to N
    Set $c = \cos(\omega_{ij,t}),\ s = \sin(\omega_{ij,t})$
    Set $t_1 = v_t[i],\ t_2 = v_t[j]$
    Set $v_t[i] \leftarrow c * t_1 - s * t_2$
    Set $v_t[j] \leftarrow s * t_1 + c * t_2$
  End For
End For
$v_t = v_t \circ \text{diag}(\Lambda_t^{-1/2})$

Algorithm 2: Recursive algorithm for computing v_t in $O(N^2)$ time; $\text{diag}(A)$ is the vector of the diagonal elements of a square matrix A.

The derivatives of the log density (4) with respect to the vector of log eigenvalues h_t is simply $-1/2 + (1/2)v_t \circ v_t$, where the symbol $\circ$ denotes element-wise product, and it is computed in $O(N)$ time given that we have pre-computed v_t. The partial derivative with respect to each rotation angle $\omega_{ij,t}$ takes the form

$$-v_t^T \frac{\partial v_t}{\partial \omega_{ij,t}} = -v_t^T \Lambda_t^{-\frac{1}{2}} \left(G_{NN-1}^T \dots G_{ij-1}^T\right) \frac{\partial G_{ij,t}^T}{\partial \omega_{ij,t}} \left(G_{ij+1}^T \dots G_{12}^T\right) r_t = -\alpha_{ij,t}^T \beta_{ij,t}$$

where $\alpha_{ij,t} = v_t^T \Lambda_t^{-1/2} (G_{NN-1}^T \dots G_{ij-1}^T)$ and $\beta_{ij,t} = (\partial G_{ij,t}^T / \partial \omega_{ij,t})(G_{ij+1}^T \dots G_{12}^T) r_t$ and the partial derivative matrix $(\partial G_{ij,t} / \partial \omega_{ij,t})$ is very sparse, having only four non-zero elements, given by

$$\frac{\partial G_{ij,t}}{\partial \omega_{ij,t}}[k,l] = \begin{cases} -\sin(\omega_{ij,t}), & \text{if } k = l = i \text{ or } k = l = j \\ \cos(\omega_{ij,t}), & \text{if } k = j, l = i \\ -\cos(\omega_{ij,t}), & \text{if } k = i, l = j \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

where $i < j$. All $\alpha_{ij,t}$ and $\beta_{ij,t}$, for $i < j$, can be computed in $O(N^2)$ time by carrying out two separate forward and backward recursions constructed similarly to the Algorithm 2. Then, all final $N(N-1)/2$

dot products $\alpha_{ij,t}^T \beta_{ij,t}$ that give the derivatives for all Givens angle parameters can be computed in overall $O(N^2)$ time by using the fact that $\beta_{ij,t}$ contains only two non-zero elements, implying that an individual dot product $\alpha_{ij,t}^T \beta_{ij,t}$ takes $O(1)$ time. This is due to the fact that the final multiplication in the computation of β is performed with the sparse matrix $\partial G_{ij,t}/\partial \omega_{ij,t}$ that has only four non-zero elements. A complete pseudo-code of the above procedure is given in the Supplementary material.

To illustrate the computational cost of the full MSV model and how it increases with T and N, in Table 1, we report the computing times in seconds (on a Windows 1.8GHz PC) per 100 MCMC draws based on a small Monte Carlo exercise for different combinations of T and N. The numbers in Table 1 are based on the average computing time (averaged over 100 replications) of fitting our full MSV model to i.i.d. standard normal artificially generated samples of different sizes. It is clear from the numbers reported in the table that computing time increases linearly in T and nonlinearly in N, as expected.

Computing times in seconds per 100 MCMC draws					
T/N	1	5	10	50	100
50	0.14	0.31	0.73	23.17	173.73
100	0.24	0.55	1.38	45.65	352.52
200	0.44	1.03	2.69	99.25	708.91
500	1.02	2.46	7.20	245.26	1766.69
1000	2.10	5.12	15.29	488.30	3495.84

Table 1: Computing times in seconds for combinations of sample sizes and model sizes

5.2 Sampling the factors in $O(TNK)$ time

The exact Gibbs step for sampling each latent factor vector f_t scales as $O(K^3)$ while sampling all of such vectors requires $O(TK^3)$ time, a cost that is prohibitive for large scale multivariate volatility datasets. To see this, notice that the posterior full conditional distribution over f_t is written in the form

$$p(f_t|\text{rest}) \propto \mathcal{N}(r_t|Bf_t, VI)\mathcal{N}(f_t|0, \Sigma_t), \tag{6}$$

which gives a density for $p(f_t|\text{rest})$ of the form $\mathcal{N}(f_t|M_t^{-1}B^T V^{-1} r_t, M_t^{-1})$ where $M_t = B^T V^{-1} B + \Sigma_t$. To simulate from this density we need to first compute the stochastic volatility matrix Σ_t and subsequently

the Cholesky decomposition of M_t. Both operations have a cost $O(K^3)$ since the matrix product $B^T V^{-1} B$, that scales as $O(NK^2)$, needs to be computed once across all time instances and therefore will not dominate the computational cost since typically $N \ll TK$. Furthermore, given that there is a separate matrix M_t for each time instance we need in total T computations of the volatility and Cholesky matrices in each iteration of the sampling algorithm, which adds a cost that scales as $O(TK^3)$. The matrix-vector products Bf_t, needed to compute the means of the Gaussians, scale overall as $O(TNK)$, but in practice this will be much less expensive than the term $O(TK^3)$. We note here that a matrix multiplication is the simplest computation with little overhead that can be trivially parallelised in modern hardware. To avoid the $O(TK^3)$ computational cost we replace the exact Gibbs step with a much faster Metropolis within Gibbs step that scales as $O(T(NK + K^2))$. Specifically, given that (6) is of the form of a latent Gaussian model, where $\mathcal{N}(f_t|0, \Sigma_t)$ is the Gaussian prior and $\mathcal{N}(r_t|Bf_t, VI)$ the (Gaussian) likelihood, we can apply the auxiliary gradient-based scheme as described in Section 4.2. By introducing the auxiliary random variable U_t drawn from

$$p(U_t|f_t) = \mathcal{N}(U_t|f_t + \frac{\zeta_t}{2} D_{f_t}, \frac{\zeta_t}{2} I),$$

where $D_{f_t} = \nabla \log \mathcal{N}(r_t|Bf_t, VI) = B^T V^{-1}(r_t - Bf_t)$, the auxiliary gradient-based method is applied as shown in Algorithm 3. Now observe that the step for sampling y takes $O(K^2)$ time because the eigenvalue

(i) $U_t \sim \mathcal{N}(U_t|f_t + (\zeta_t/2)D_{f_t}, (\zeta_t/2)I)$
(ii) Propose $y \sim \mathcal{N}(y|(2/\zeta_t I + \Sigma_t^{-1})^{-1}(2/\zeta_t)U_t, ((2/\zeta_t)I + \Sigma_t^{-1})^{-1})$ and accept it with probability
$r = \frac{\mathcal{N}(r_t|By,VI)}{\mathcal{N}(r_t|Bf_t,VI)} \exp\Big\{-(U_t - f_t)^T D_{f_t} + (U_t - y)^T D_y - \frac{\zeta_t}{4}(||D_y||^2 - ||D_{f_t}||^2)\Big\}$.

Algorithm 3: Auxiliary gradient-based sampling for the latent factors

decomposition of the covariance matrix $((2/\zeta_t)I + \Sigma_t^{-1})^{-1}$ can be expressed analytically as $P_t((2/\zeta_t)I + \Lambda_t^{-1})^{-1} P_t^T$ where $\Sigma_t = P_t \Lambda_t P_t^T$ is the spectral decomposition of Σ_t. Therefore, we can essentially apply Algorithm 2 to sample y in $O(K^2)$ time. Furthermore, the most expensive operation in the M-H ratio above is the computation of the matrix-vector product Bf_t which costs $O(NK)$. Therefore, overall for all factors across time we need $O(T(NK+K^2))$ operations which is typically dominated by $O(TNK)$ since $K \ll N$.

Thus, by taking advantage of the analytic form of the eigenvectors of Σ_t, the computational complexity for proposing y (when sampling f_t) is reduced from $O(K^3)$ to $O(K^2)$. The crucial sharing of eigenvectors property is due to the spherical or isotropic covariance matrix $(2/\zeta)I$ that is added to Σ_t^{-1} and it does not alter the eigenvectors. In contrast, this does not hold for the Gaussian proposal used in Gibbs sampling because the inverse covariance matrix of that proposal, given by $M_t = B^T V^{-1} B + \Sigma_t^{-1}$, is obtained by adding an non-isotropic covariance matrix $B^T V^{-1} B$ to the matrix Σ_t^{-1} which makes the eigenvectors of M_t different from those of Σ_t.

5.3 Full factor MSV model against an MSV model

From a computational perspective, the full factor model has an advantage over the model in (1), because it deals with missing values more efficiently. Suppose that $N = K$ and that we use the multivariate model in (1) and the returns vector at time t contains missing values, so that $r_t = (r_{t,o}, r_{t,m})$ where $r_{t,o}$ is the sub-vector of observed components and $r_{t,m}$ are the missing components. The standard Bayesian treatment is to marginalize out the unobserved values $r_{t,m}$ and obtain the likelihood term (at time t) given by $r_{t,o} \sim \mathcal{N}(0, \Sigma_{t,o})$ where $\Sigma_{t,o}$ is the sub-block of the full covariance matrix Σ_t that corresponds to the observed dimensions. The computation of $\Sigma_{t,o}$ is expensive since it requires first the computation and storage of the full matrix Σ_t which scales as $O(N^3)$. Given that we have T such matrices the whole computation scales as $O(TN^3)$ which is prohibitively expensive, especially within an MCMC algorithm where these computations are repeated in each iteration. A Gibbs step for sampling the missing $r_{t,m}$, instead of marginalizing them out, also suffers from the same computational cost. In contrast, for the full factor model the treatment of missing values is very simple since $r_{t,o}$ and $r_{t,m}$ are conditionally independent given the latent factors f_t and thus the marginalization of $r_{t,m}$ is trivial. The computational burden is moved to the MCMC sampling of the latent vectors f_t, but as we showed in Section 5.2, sampling f_t can be achieved in $O(TNK)$ time.

Since in nearly all financial applications of daily asset returns there are many missing values, encountered for example because in multinational market portfolios holidays differ between countries, we recommend using the full factor model with $N = K$, $B = I$ and $v_i > 0$ for all i even when the number of assets

N is manageable.

5.4 Comparison with other approaches

5.4.1 A computational strategy based on sequential Monte Carlo

Another popular computational strategy to implement sequential analysis of state-space models such as our proposed MSV model is sequential Monte Carlo, see for example Chopin and Papaspiliopoulos (2020). The main strength of these methods is that they deliver attractive theoretical properties. Traditionally, their main disadvantage has been that their computational efficiency deteriorates exponentially with the dimension of the latent space, so their performance can be poor in large-dimensional systems; see for example Rebeschini and Van Handel (2015). However, recently there has been a lot of work that has delivered operational improvements with low computational cost in moderate to high dimensions. In this subsection, we investigate the option of applying an SMC computational strategy for inference in our MSV model.

We adopt the SMC^2 approach (Chopin et al., 2013; Fulop and Li, 2013) in which two set of particles are carried forward, one for the latent variables and one for the static parameters. We perform a small experimental study that compares our MCMC algorithm with SMC^2. We use subsets of the first $T = 100$ daily returns of the first $N = 8$ stocks from the asset returns dataset from Section 6.2 (detailed information on the dataset can be found in Section 6.2.1) and we perform SMC^2 with the default settings as suggested in the accompanying Python package of Chopin and Papaspiliopoulos (2020). Table 2 compares minimum effective sample sizes (ESS) per running times (seconds) of SMC^2 and our MSV algorithm. We find that SMC^2 performs favorably except when both T and N increase.

	$T = 25$		$T = 50$		$T = 100$	
	MCMC	SMC^2	MCMC	SMC^2	MCMC	SMC^2
$N = 3$	23.1	17.6	8.6	9.2	3.6	4.4
$N = 4$	15.3	10.4	6.3	4.7	2.2	3.6
$N = 5$	9.3	7.0	3.1	4.1	1.0	0.05
$N = 6$	6.0	5.4	2.2	3.2	0.14	0.09
$N = 7$	4.2	4.1	0.6	2.0	0.1	0.005
$N = 8$	1.2	2.8	0.9	0.7	0.1	0.005

Table 2: Minimum Effective sample size per second for MCMC and SMC^2 based inference for an N-dimensional MSV model for time slots T.

It is possible to further improve the performance of the naive SMC^2 implementation above by adopting modern implementation strategies, some of which we outline here. The algorithm is parallel in the static parameter dimension (Duan and Fulop, 2015) and it can benefit from the massive computational power of GPU parallel architectures (Fulop and Li, 2013, 2019). For fixed ESS, Chopin et al. (2013) state that the computational complexity of SMC^2 is $O(N_\theta T^2)$ where N_θ denotes the number of particles for the static parameters of the model. In practice this means that, with P parallel processors the same ESS can be achieved with the same computational cost for $\sqrt{P}T$ instead of T data points. A major improvement may also be achieved by employing a sequential quasi-Monte Carlo strategy (Gerber and Chopin, 2015) that achieves a $O(N_\theta T)$ rate of convergence. Finally, a simple idea (Chopin and Papaspiliopoulos, 2020, Section 18.3) that may dramatically improve the efficiency of SMC^2 is based on the observation that it seems preferable to run many SMC^2 replicates with lower N_θ and then average the results. Chopin and Papaspiliopoulos (2020) report that 25 replications of the algorithm with $N_\theta = 10^3$ produce the same level of variance reduction as with one run with $N_\theta = 10^4$. By additionally exploiting 25 parallel cores, this would imply that this simple method has the potential to achieve the same variance reduction in data size five times larger. Thus, our conclusion is that SMC^2 implementation accompanied with a selection of the above strategies can perform better than MCMC for moderate values of N and T. The comparative merits of the two algorithms in high dimensions and large data sizes requires further study and it is beyond the scope of this paper.

5.4.2 Comparison with a Wishart autoregressive process

Another modelling approach in multivariate stochastic volatility models that has recently attracted the attention of the machine learning community is the Wishart autoregressive (WAR) process introduced by Gouriéroux et al. (2009). They provide an alternative interpretation of a WAR process of order one, WAR(1) as an outer product of Gaussian autoregressive vector processes as follows. Suppose that $u_{ij}(t)$ for $i = 1, \ldots, \nu$ and $j = 1, \ldots, N$ are zero-mean Gaussian processes with an exponential kernel function

$k(t,s) = \exp(-\lambda|t-s|)$ with $\lambda > 0$. Then

$$\Sigma_t = \sum_{i=1}^{\nu} L u_i(t) u_i(t)^T L^T \sim W_D(V, \nu)$$

where $u_i(t) = (u_{i1}(t), \ldots, u_{iN})^T$, L is the Cholesky decomposition of the $N \times N$ matrix V and $W_D(V, \nu)$ denotes the Wishart density with scale matrix V and degrees of freedom ν with $\nu > N$.

Wilson and Ghahramani (2010) generalise the WAR processes by varying the kernel of the Gaussian process and propose an MCMC algorithm to estimate the parameters u and L when some observed time series r_t are modelled as N-dimensional zero mean, normally distributed, time series vectors with covariances Σ_t. They also suggest setting $\nu = N + 1$ which they found to be effective in their 3-dimensional WAR(1) multivariate stochastic volatility dataset applied on exchange rate returns.

The most serious drawback of the Wishart process approach is that it suffers from the usual $O(N^3)$ cubic complexity due to the need to invert the matrix Σ_t to evaluate the likelihood function. Thus, Wishart process models are not able to scale well with large N. To provide an illustration on the resulting covariance matrix estimates of the WAR and how they compare to our MSV approach, we use the daily dataset of Section 6.2 (detailed information on the dataset can be found in Section 6.2.1) and perform Bayesian inference for the WAR(1) model by applying a Hamiltonian Monte Carlo sampler (Girolami and Calderhead, 2011). A small subset based on the first $N = 3$ stocks and $T = 50$ data points of our daily returns dataset comprising of $N = 571$ and $T = 2017$ took about 11 hours to run 10,000 iterations (5,000 warm-up) in PyStan (which uses the fast HMC algorithm written in C++) (Carpenter et al., 2017) on a Windows 1.8GHz PC whereas our MSV model ran in 78 seconds.

In addition to the computational complexity, a notable difference between the two models is that the WAR(1) model has $\nu N = N(N+1) = 12$ latent autoregressive processes whereas the MSV model has $N(N+1)/2 = 6$. As a result, a small dataset of $T = 50$ does not allow the Wishart model to fit the data adequately, producing smooth estimates that are dominated by the smoothing parameters priors. We use $\lambda \sim Ga(1,1)$, a LKJ(2)-correlation prior (Lewandowski et al., 2009) for the lower part of L_D and a $\mathcal{N}(0,10)$ for the diagonal elements of L_D. Figure 1 presents the data and the posterior mean estimates of MSV and

WAR(1) models. The smoothness in the WAR(1) estimates is evident when a 'jump' appears in the volatility of returns: the MSV models responds faster. We expect that this phenomenon will be alleviated when T increases but the computational complexity of the WAR(1) model makes such comparison infeasible.

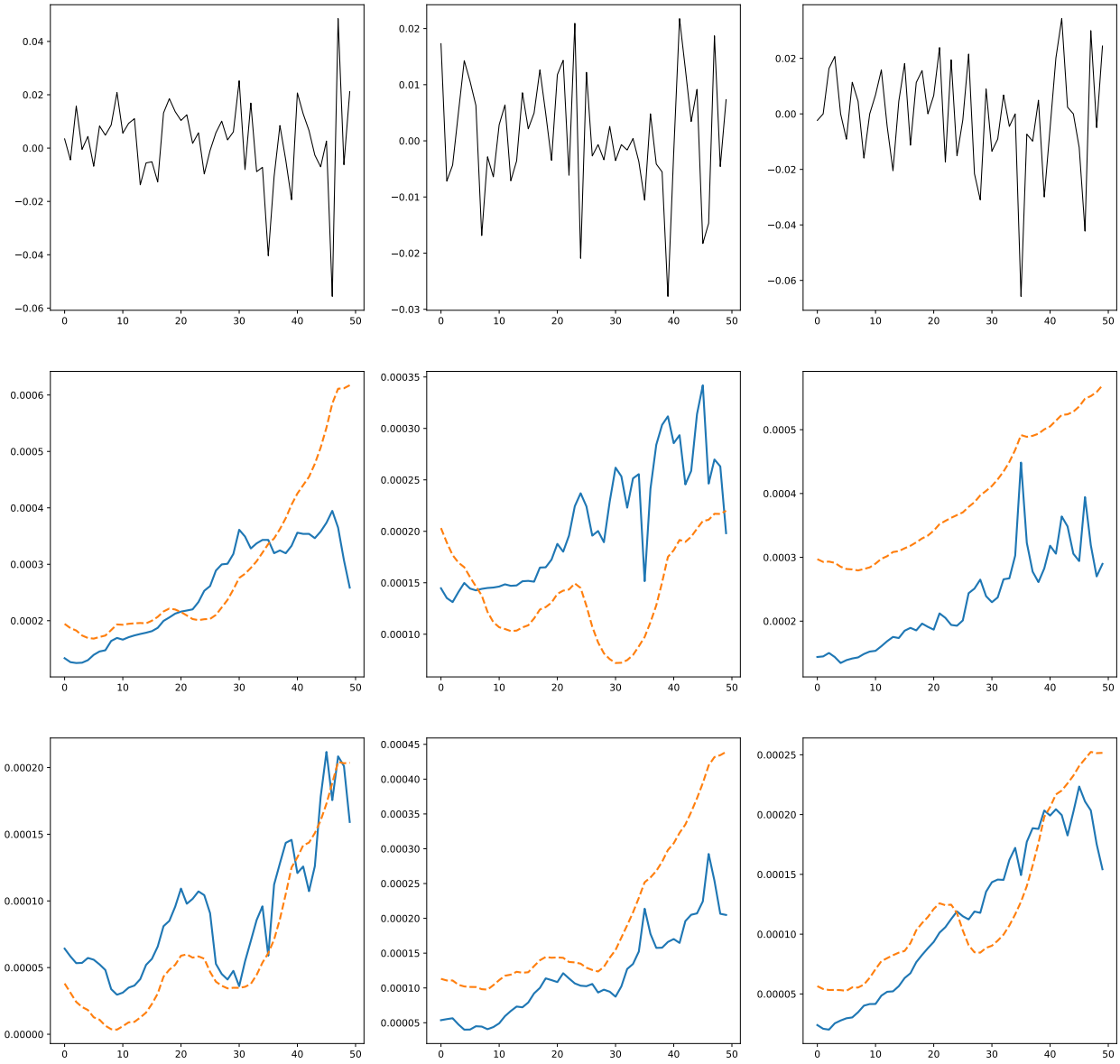

Figure 1: Data (first row) and posterior means of variances (second row) and covariances (third row) for 3 stock returns in 50 time slots. Blue solid line: MSV model; Orange dashed line: WAR(1) model.

6 Applications

6.1 Small-dimensional macroeconomic application

6.1.1 The stochastic volatility VAR model

Stochastic volatility models have received a lot of attention in the applied macroeconomic literature, due to their ability to accommodate structural changes or breaks in the volatility of macroeconomic series. There has been considerable evidence (Sims (1980), Bernanke and Mihov (1998), Kim and Nelson (1999), McConnell and Perez Quiros (2000), Sims and Zha (2006), Primiceri (2005)) documenting that inflation and unemployment in the U.S. have undergone a period of high volatility during the oil shocks in the late 1970s and early 1980s, followed by an exceptionally low volatility period 1984-2000 known as the Great Moderation. Moreover, the increase in financial volatility during the 2007-2008 financial crisis has generated renewed interest in the estimation of changes in macroeconomic volatility.

The standard approach to the estimation of time-varying volatility in the macroeconomic literature has been to use multivariate time series models such as vector autoregressions (VARs) with stochastic volatility. This approach allows for joint dynamics of the series while modelling the stochastic volatility in a multivariate fashion also permits the covariances between the series to evolve over time. It is straightforward to employ the MSV model of Section 2 or its factor extension to the residuals of a VAR(p) model

$$y_t = B_0 + \sum_{i=1}^{p} B_i y_{t-i} + \varepsilon_t, \;\; \varepsilon_t | \Sigma_t \sim \mathcal{N}(0_M, \Sigma_t). \tag{7}$$

The standard approach in the applied macroeconomic literature (e.g. Primiceri (2005)) for maintaining symmetry and positive definiteness of the stochastic covariance matrix has been to use a Cholesky decomposition of the form $\Sigma_t = A_t^{-1} \Lambda_t A_t^{-1\prime}$, assuming that the elements of the diagonal Λ_t follow random walk in logarithms[1], and modelling the elements of the inverse lower unitriangular matrix in the Cholesky decomposition A_t^{-1} as random walks. This approach has become prevailing in the literature with vast work applying

[1]To sample from the geometric random walks, Primiceri (2005) employs a procedure by Kim et al. (1998b) and Cogley and Sargent (2005) use a Metropolis within Gibbs step from Jacquier et al. (1994a).

the algorithms from Cogley and Sargent (2005) and Primiceri (2005) to different macroeconomic problems. One of the drawbacks of this approach is that the ordering of the variables in the VAR model matters for inference. The triangularisation of the covariance matrix is useful for conducting structural analysis as in Primiceri (2005), where the ordering of the variables serves as an identifying restriction for the monetary policy shock. However, in models where structural shock analysis is not required and the reduced-form VAR model is used for estimation or forecasting, diagonalisation has only been employed as means to facilitate estimation of the drifting volatilities (Cogley and Sargent (2005), Clark (2012), Koop and Korobilis (2013), Carriero et al. (2016), Koop and Korobilis (2019)) and has the undesirable effect of making the model-implied estimates and forecasts depend on the ordering of the variables in the system. To make this clear, suppose that we have an $M \times 1$ dimensional vector y_t generated by the stochastic volatility VAR(p) model in (7). Then the typical element of Σ_t is given by $\Sigma_{ij,t} = \sum_{k=1}^{M} \lambda_{kk,t}\tilde{a}_{jk,t}\tilde{a}_{ik,t}$, where $\lambda_{kk,t}$ is the k^{th} diagonal element of Λ_t, and $\tilde{a}_{jk,t}$ is the jk^{th} element of the lower unitriangular A_t^{-1}. For a system of three variables, for example, this implies the following parametric restrictions on Σ_t :

$$\Sigma_t = \begin{bmatrix} \lambda_{11,t} & \lambda_{11,t}\tilde{a}_{21,t} & \lambda_{11,t}\tilde{a}_{31,t} \\ \lambda_{11,t}\tilde{a}_{21,t} & \lambda_{11,t}\tilde{a}_{21,t}^2 + \lambda_{22,t} & \lambda_{22,t}\tilde{a}_{32,t} + \lambda_{11,t}\tilde{a}_{21,t}\tilde{a}_{31,t} \\ \lambda_{11,t}\tilde{a}_{31,t} & \lambda_{22,t}\tilde{a}_{32,t} + \lambda_{11,t}\tilde{a}_{21,t}\tilde{a}_{31,t} & \lambda_{11,t}\tilde{a}_{31,t}^2 + \lambda_{22,t}\tilde{a}_{32,t}^2 + \lambda_{33,t} \end{bmatrix}.$$

The order of the variables in the system together with the assumptions on the stochastic law of motion and the distribution of the elements of A_t and Λ_t therefore have an implication on the stochastic properties of the elements of Σ_t. For example, when geometric random walk process with Gaussian errors is assumed for the elements of Λ_t and Gaussian random walks are assumed for the elements of A_t^{-1}, the stochastic volatility of the first variable in the system, $\Sigma_{11,t}$, is log-Normal, while the stochastic volatility of the second variable involves multiplicative terms between elements of Λ_t and quadratic terms of the elements of the inverse of A_t whose elements follow Gaussian random walks. In general, there is $M!$ number of possible permutations, which can be extremely large even for small M. For example, in Koop and Korobilis (2013) $M = 25$, and there are over 15 quadrillion possible orderings (1.55×10^{25}) which could potentially lead

to very different conclusions, and it is clearly computationally infeasible to check all possible permutations. In contrast, our MSV specification on Σ_t is invariant to ordering of the variables in the system, and hence robust to such permutations.

To illustrate how the order of the variables can lead to empirically different conclusions, we estimate a VAR model on U.S. unemployment, inflation and nominal interest rate for the period 1948Q1-2019Q2, using the approach of Primiceri (2005) and report the estimated pairwise covariances and correlations between the three variables for the six possible different orderings. Figure 2 presents the estimated paths of the off-diagonal elements of Σ_t, the top panel displays the estimates of a constant-parameter VAR(2) with our multivariate stochastic volatility (MSV) specification, the bottom panel displays selected orderings of a constant-parameter VAR(2) with volatility specification as in Primiceri (2005), and the dotted lines show the 68% credible sets (the Supplementary material contains the estimates of all six orderings). It is clear from Figure 2 that changing the order of the variables in the VAR can result in completely different paths for the estimated covariances over time: when inflation is ordered last, after unemployment and interest rate, the estimated covariance between unemployment and inflation is positive around the 1980s, while when inflation is ordered second, after interest rate and before unemployment, the estimated covariance between unemployment and inflation is negative at the same period. Similar discrepancies can be found when looking at the estimates under various permutations for the covariance between the nominal interest rate and inflation, or unemployment respectively.

Figure 3 further reiterates the point by displaying the estimated paths of the pairwise correlations (taking into account the time variation in the diagonal elements), the top panel displays the estimates of a constant-parameter VAR(2) with our dynamic MSV specification, the bottom panel displays selected orderings of a constant-parameter VAR(2) with volatility specification as in Primiceri (2005) (the Supplementary material contains the estimates of all six orderings).

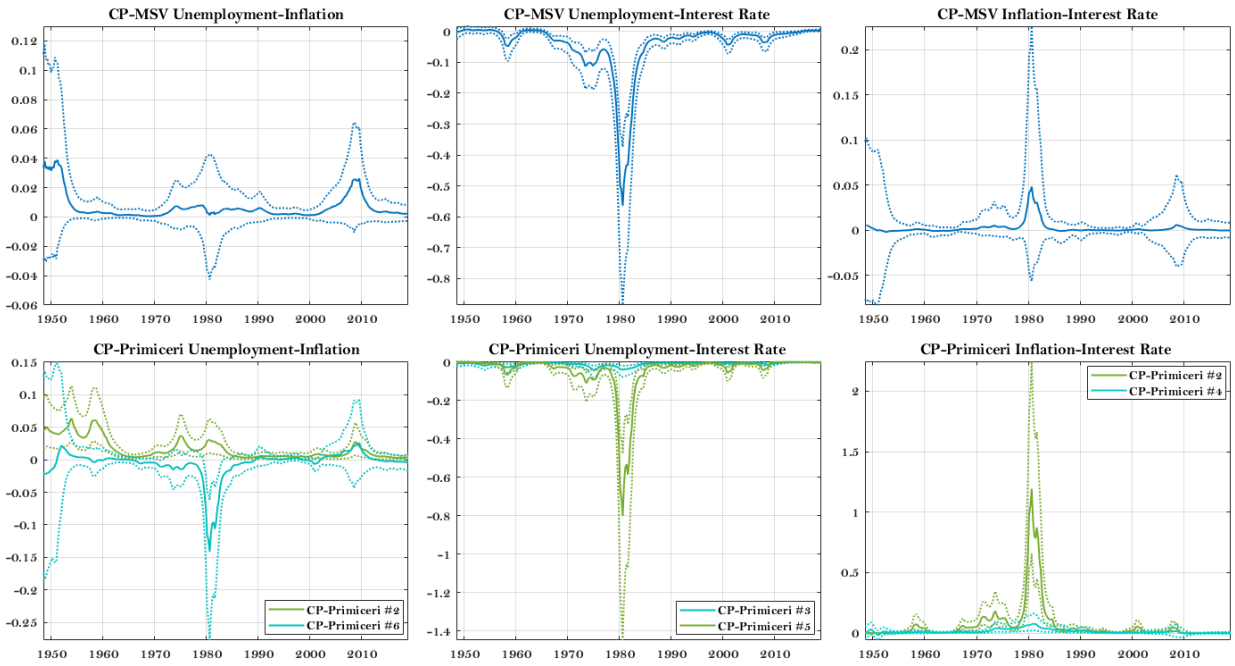

Figure 2: Estimated off-diagonal elements of the covariance matrix

6.1.2 Extension to time-varying autoregressive parameter VAR

Next, we consider further allowing the autoregressive parameters to change over time. For our MSV specification, we estimate the time variation in the parameters, using the kernel-weighted quasi-Bayesian approach of Petrova (2019), which models the drifts in the autoregressive matrix nonparametrically with the use of a kernel (additional details on the extra Gibbs step in the MCMC can be found in the Supplementary material). For the alternative specification, we use Cholesky volatility and random walk processes for the autoregressive parameters as in Primiceri (2005). Figures 4 and 5 present the estimated paths for the covariances and pairwise correlations respectively after allowing for time variation in the autoregressive parameters, for our the time-varying parameter TVP-MSV model and for selected permutations of the TVP Primiceri (2005) model (the Supplementary material contains the estimates of all six orderings). It is clear that even after allowing for time-variation in the autoregressive matrices and hence in the conditional mean of the series, the uncovered discrepancies for different permutations remain and different decisions on the ordering of the

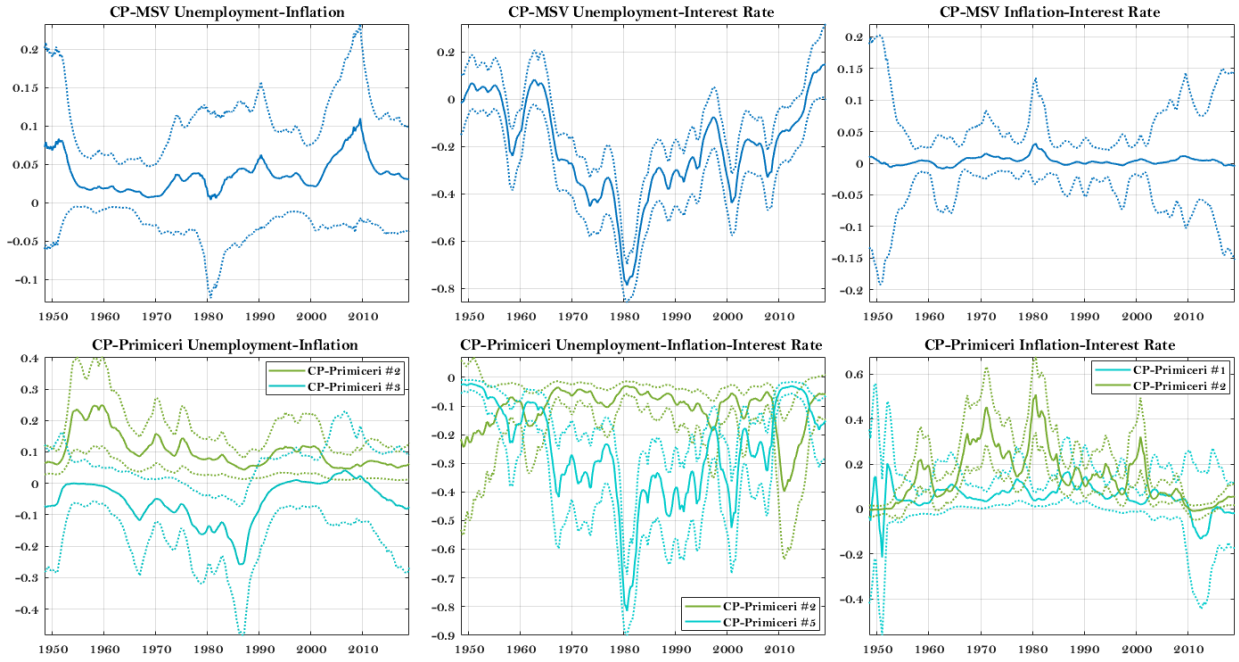

Figure 3: Estimated pair-wise correlations

variables can lead to different empirical conclusions.

6.1.3 The long-run volatility and inflation persistence

Next, we estimate the long-run volatility of the three series, computed as $\sum_{j=0}^{\infty} C_t^j \Sigma_t C_t^{j\prime}$, where C_t is the corresponding time-varying companion matrix of the VAR model. The top panel of Figure 6 presents the volatility estimates with our TVP-MSV VAR model, and the bottom panel presents the estimates of the TVP Cholesky specification with the variables ordered as in Primiceri (2005) (unemployment, inflation and the nominal interest rate).

From Figure 6, there are several interesting results. First, for the estimated long-run volatility of unemployment using our TVP-MSV model, we find that the spike in volatility during the 2008 financial crisis is much larger compared to the estimated volatility using the Cholesky specification. Interestingly, it is also larger than the spike estimated with our model during the oil crises periods. Second, our estimated long-run interest rate volatility is much lower during the period of Paul Volcker as chairman of the Fed than

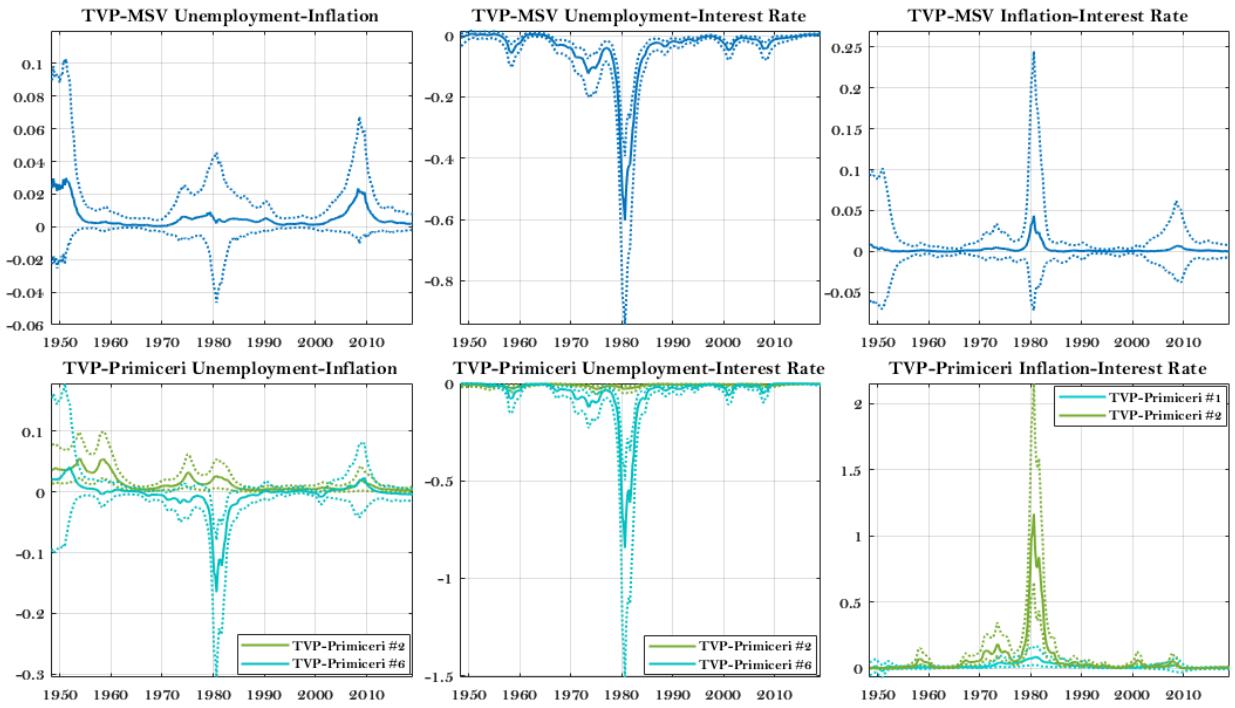

Figure 4: Estimated off-diagonal elements of the covariance matrix

the estimates with the Cholesky specification of the model, which we think is reasonable particularly since the long-run volatility can be viewed as the volatility of the series when an infinite horizon in the future is considered. The large difference in the estimated volatilities is partly a consequence of the difference in the modelling choices for the autoregressive parameters, since the random walk specification proposed by Cogley and Sargent (2005) delivers draws for the autoregressive matrix with spectral radius close to the unit circle (even when rejection sampling is employed to discard unstable draws). But, crucially, the main difference is due to the different modelling approaches for the multivariate stochastic volatility.

Finally, we compute the Cogley et al. (2010)'s measure of inflation persistence h steps given by

$$R^2_{t,h} = 1 - \frac{e_\pi \left[\sum_{j=0}^{h-1} C_t^j \Sigma_t C_t^{j\prime}\right] e_\pi'}{e_\pi \left[\sum_{j=0}^{\infty} C_t^j \Sigma_t C_t^{j\prime}\right] e_\pi'},$$

where e_π is a selection vector which select the inflation series. $R^2_{t,h}$ measures the proportion of total vari-

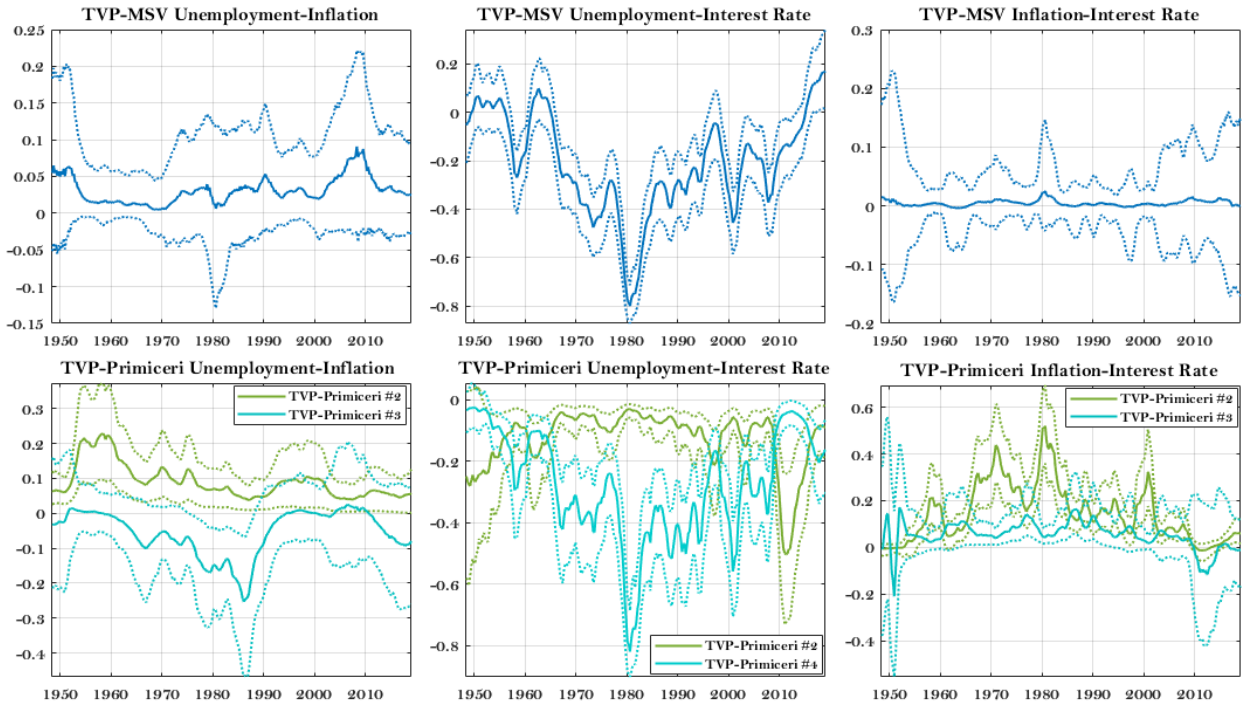

Figure 5: Estimated pair-wise correlations

ation explained by past shocks, with values close to one implying that past shocks die out slowly implying that inflation was more persistent and predictable. Figure 7 presents the posterior median of $R^2_{t,h}$ and the corresponding 68% credible sets, computed at each point in time and for horizons of one quarter, two and a half years, and five years ahead for our TVP-MSV model and the TVP Cholesky specification with ordering as in Cogley et al. (2010).

The main finding from Figure 7 is that we find a larger fall in inflation persistence after the beginning of the Great Moderation if we use the MSV specification than if we use the Primiceri (2005)'s specification for the volatility at short horizons. On the other hand, our estimate of the $R^2_{t,h}$ dies much quicker over the horizon, implying that inflation is virtually unpredictable five years in the future over all periods of the sample.

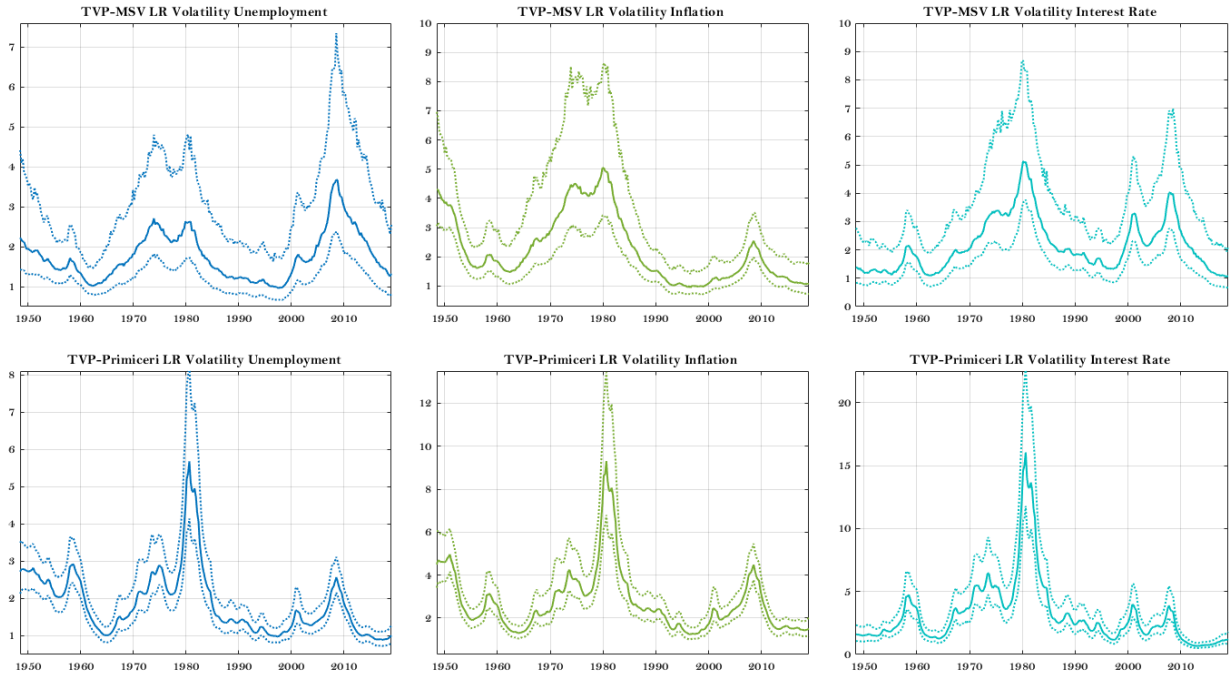

Figure 6: Long-run volatility estimates

6.2 Large-dimensional financial application to volatility of asset returns

6.2.1 Data and MCMC specifics

In this Section, we illustrate how to apply our methodology to modelling daily returns from 600 stocks of the STOXX Europe 600 Index downloaded from Bloomberg between 10/1/2007 to 5/11/2014. The cleaning of the data involved removing 29 stocks by requiring for each stock at least 1000 traded days and no more than 10 consecutive days with unchanged price. The final dataset has $N = 571$ and $T = 2017$. There were 36340 missing values in the data due to non-traded days, asynchronous national holidays, etc. A smaller dataset was created by selecting 1000 days and the 29 stocks from the Italian stock market. This dataset was used for a large empirical study to compare our models with independent factor MSV models.

All MCMC algorithms in this Section has an adaptive time for burn-in consisted of 10^4 iterations with resulting acceptance probability of $50 - 60\%$ for all auxiliary Langevin steps. After this adaptive burn-in phase all proposal distributions are kept fixed and then we further perform 10^4 iterations to finally collect

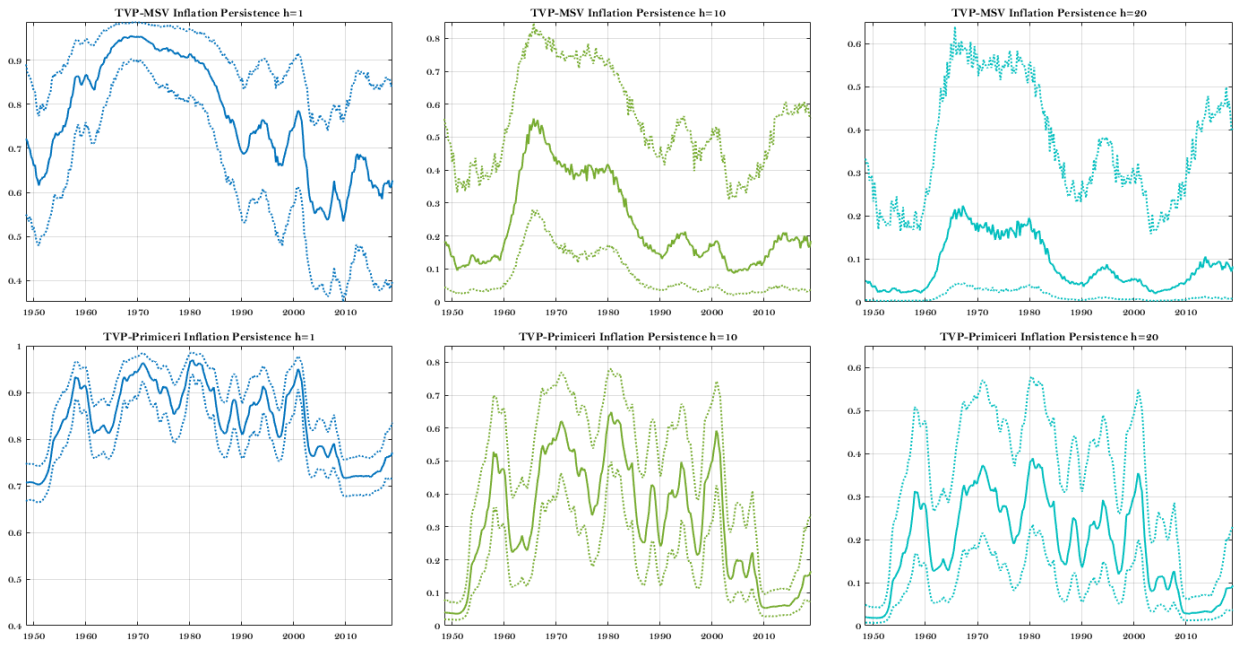

Figure 7: Inflation Persistence

10^3 or 2×10^3 (thinned) samples for the large and small datasets respectively.

6.2.2 Predictive ability

We compare the predictive ability of our full factor MSV model against an independent factor MSV model in which all factors are assumed to be uncorrelated. The basis of our comparison is the predictive likelihood function. Each model produces a predictive distribution and therefore a predictive likelihood can be evaluated for a future observation. The comparison of these predictive likelihoods decomposes the Bayes factor one observation at a time and a cumulative predictive Bayes factor through a future time period, which serves as a Bayesian evidence in favour of a model based on predictive performance; see, for example, Geweke and Amisano (2010). Denote by X_t the dynamic latent path and by θ all the static parameters of the model. Assume that at time T we have obtained MCMC samples $X^s_{1:T}$ and θ^s, $s = 1, 2, \ldots, S$ based on observed data $r_{1:T} = (r_1, r_2, \ldots, r_T)$. For each model, the one-step-ahead predictive likelihood conditional on θ is

given by

$$p(r_{T+1}|r_{1:T},\theta) = \int p(r_{T+1}|r_{1:T},\theta)dF(X_{T+1} \mid r_{1:T},\theta)$$

and a Monte Carlo estimate is obtained as

$$\hat{p}(r_{T+1}|r_{1:T},\theta) = S^{-1}\sum_{s=1}^{S} p(r_{T+1}|X_{T+1}^{s},\theta)$$

where X_{T+1}^{s} are samples from the transition densities (1) and θ is fixed at the posterior mean estimate at time T. This procedure is repeated producing M one-step-ahead predictive likelihoods for data $r_{T+1:T+M}$ where θ is kept fixed at the sample mean $S^{-1}\sum_{i=1}^{S}\theta^{s}$ and the required samples from the density $f(X_t \mid r_{1:t}, X_{1:t-1})$ for $t = T+1,\ldots,T+M-1$ are obtained through the auxiliary particle filter of Pitt and Shephard (1999b); see Carvalho et al. (2010) for a review on particle learning. Thus, marginal likelihoods for each model can be obtained through

$$\hat{p}(r_{1:T+M}|r_{1:T},\theta) = \prod_{t=T+1}^{T+M}\hat{p}(r_{t+1}|r_{1:t},\theta)$$

and the predictive Bayes factors in favor of one model against the other can be readily calculated, see Geweke and Amisano (2010), Pitt and Shephard (1999a).

The smaller dataset based on 29 stocks was selected by excluding the last 100 days of the larger dataset such that $T = 1000$ and $M = 100$, representing a predictive period of about 4 months. Marginal likelihoods are calculated for all full and independent factor MSV models. For the class of independent factor MSV models the best model turned out to be the 2-factor model with estimated log-Bayes factor 50.9 against the second best model with 3 factors. A log-Bayes factor greater than 5 is considered to be a very strong evidence in favour of one model against the other, see Kass and Raftery (1995). Actually the marginal likelihood decreased with the number of factors, implying that the dynamic nature of the covariance structure cannot be captured by increasing the size of independent latent processes, see Figure 8. This also reflects the inherent limitation of the independent factor models that attempt to estimate dynamic correlations through

(static) linear combinations of univariate independent stochastic processes. More latent processes just add further noise resulting to a fall in the Bayes factors.

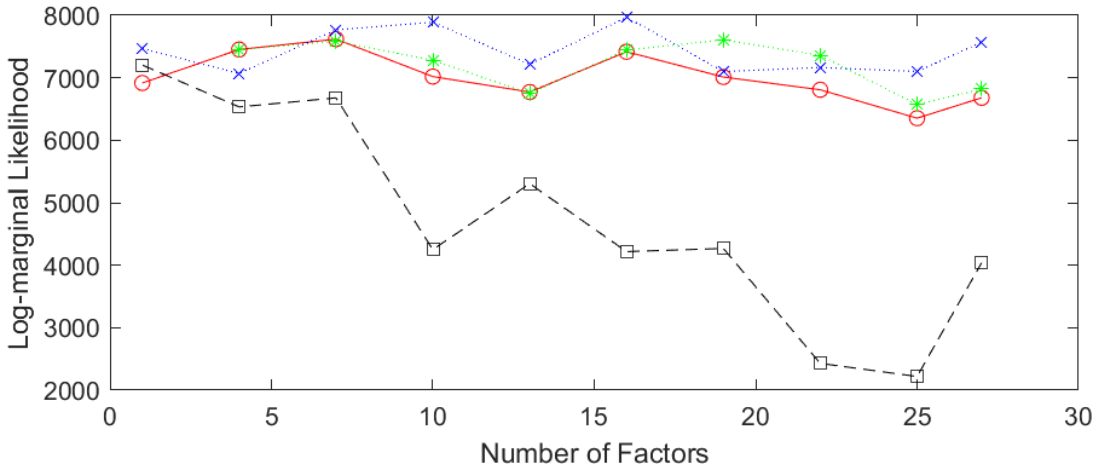

Figure 8: Logarithm of marginal likelihood. Dotted, blue, cross: full factor model with exchangeable priors; Dotted, green, star: full factor model with exchangeable priors and $\sigma_i = \sigma$; Solid, red, circle: full factor model with independent priors; Dashed, black, square: Independent factor model.

The best full factor MSV model turned out to be the 7-factor model with log-Bayes factor 385.24 against the independent 2-factor MSV model. The largest 29-factor model had a log-Bayes factor of 71.10 in favour against the best independent factor model. There is overwhelming evidence that full factor models provide better predictions. Moreover, the general picture of Figure 8 indicates that the full factor MSV model is more robust, across number of factors, when compared with the independent factor model.

Figure 8 also depicts the log-marginal likelihood of the full factor model in which, as a robustness check, independent $N(0, 10^3)$ priors were used for $\tilde{\phi}_i^h | \mu_h, \lambda_h$ instead of the exchangeable priors in (2) and the full factor model in which the elements v_i of matrix V are assumed to be equal to a fixed variance σ^2. It seems that both the exchangeability assumption and allowing for different error variances v_i achieve some predictive ability compared with a plain MSV model with independent priors and equal variances $v_i = \sigma^2$ with all our specifications performing considerably better than the independent factor model.

6.2.3 Computational efficiency

One may question whether the increased efficiency achieved by sampling the factors with the Metropolis sampler of Section 5.2 achieves a realistically faster algorithm than the simple Gibbs sampling algorithm. Figure 9 presents how computation time scales with number of factors in the large dataset. The left panel illustrates that for large problems the Gibbs sample has a prohibitive computational cost, whereas the right panel demonstrates how the computing time ratio increases with the number of factors.

In smaller examples with few factors in which the computation of the Gibbs sampler is feasible, it is of interest to inspect the Markov chain mixing of the two algorithms. For the small dataset, Table 6.2.3 presents the effective sample sizes and the computing times for the 7-factor full MSV model with the Metropolis and Gibbs samplers. The parameters inspected are the $29 \times 30/2$ elements of the 29-dimensional covariance matrix Σ_T based on $10,000$ unthinned iterations. The computing times are comparable, but the Metropolis algorithm clearly outperforms the Gibbs sampler in terms of Markov chain convergence efficiency.

Method	Time(s)	Minimum ESS	Maximum ESS	s / Minimum ESS
Metropolis	2954.2	3025.3	6538.7	0.98
Gibbs	3339.2	122.2	4164.3	27.33

Table 3: Effective sample sizes (ESS) and computing times in seconds (s) for sampling the factors in the 7-factor full MSV model

6.2.4 Results from the large dataset

To apply our full factor MSV model to the large dataset we need to choose the number of factors K. The scale of the problem makes the calculation of marginal likelihoods for each K computationally infeasible, so we propose comparing one-step ahead forecasts of different K against a proxy. We use as a proxy for Σ_{T+1} the realized covariation matrix calculated as the cumulative cross-products of five minutes intraday returns; see Andersen et al. (1999) and Barndorff-Nielsen and Shephard (2004). If an element of an $N \times N$ covariance matrix σ_{ij} is estimated by the elements of the posterior mean of Σ_{T+1} with elements $\hat{\sigma}_{ij}$ and its corresponding proxy estimate is σ^*_{ij}, we use as discrepancy measures to test how competing models perform the mean absolute deviation given as $N^{-2} \sum_{i,j} |\sigma^*_{ij} - \sigma_{ij}|$ and the root mean square error given by

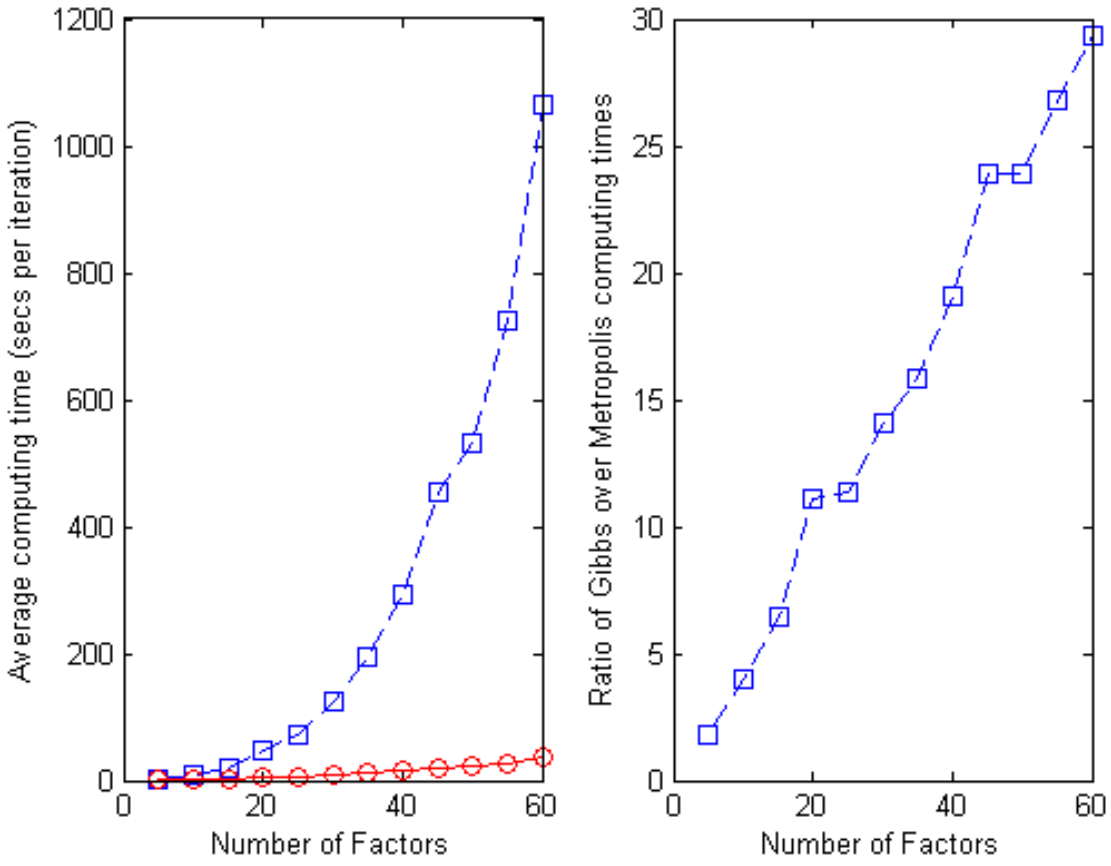

Figure 9: Left panel: Average computing time for Metropolis-Hastings (circle) and Gibbs sampling (square) algorithms. Right panel: Computing time ratio between Gibbs sampling and Metropolis-Hastings algorithms.

$[N^{-2}\sum_{i,j}(\sigma^*_{ij}-\sigma_{ij})^2]^{1/2}$.

For $K = 20, 30$ and 40 the corresponding values of these quantities were $(0.0605, 0.0601, 0.0635)$ and $(0.0567, 0.0553, 0.0614)$ respectively, so there is an indication that out of sample forecasting ability of the STOXX 600 volatility matrix is better with around $K = 30$ factors. Sometimes prediction of more days ahead might be of interest, for example when portfolio re-allocation is performed in different time scales, so we also predicted Σ_{T+2} and again the corresponding discrepancy measures were $(0.0763, 0.0720, 0.0811)$ and $(0.0775, 0.0697, 0.0867)$ respectively, verifying that $K = 30$ factors have a comparatively better pre-

dictive ability.

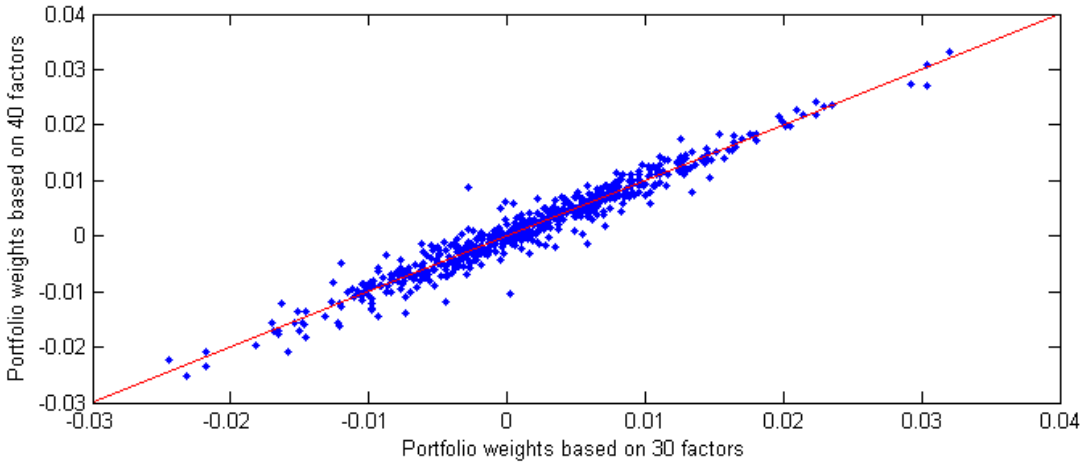

Figure 10: Next day minimum variance portfolio weights of 571 stocks of *STOXX Europe 600 index* based on 30 factors against those based on 40 factors.

Figure 10 presents the 571 minimum variance portfolio weights with 30 and 40 factors calculated as $\Sigma_{T+1}^{-1}\iota/\iota'\Sigma_{T+1}^{-1}\iota$ where Σ_{T+1} is estimated with the MCMC-based posterior predictive mean and ι is an $N\times 1$ vector of ones. It is clear that the magnitude of the weights remains considerably constant especially in the financially important values away from zero. By updating the parameters of our model on a daily basis, this minimum variance portfolio produced, over 10 consecutive days, mean daily returns and standard deviations 0.0016 (0.0056), 0.0017 (0.0044) and 0.0013 (0.0045) for the 20, 30 and 40 factor models respectively. The equally-weighted portfolio obtained smaller variance with corresponding values 0.0015 (0.0078).

Figures 11 and 12 are image plots of all estimated daily pairwise $571 \times 570/2$ correlations and 571 variances of all stocks across the whole period under study. It is interesting that these graphs allow visual inspection of European financial contagion events by inspecting, vertically, simultaneous correlation and volatility increases. Indeed, it is clear that our model has identified the early 2009 financial crisis with events such as plummeting of UK banking shares, all-time high number of UK bankruptcies and eight U.S. bank failures. Moreover, one can see the mid-2012 crisis after a scandal in which Barclays bank tried to manipulate the Libor and Euribor interest rates systems.

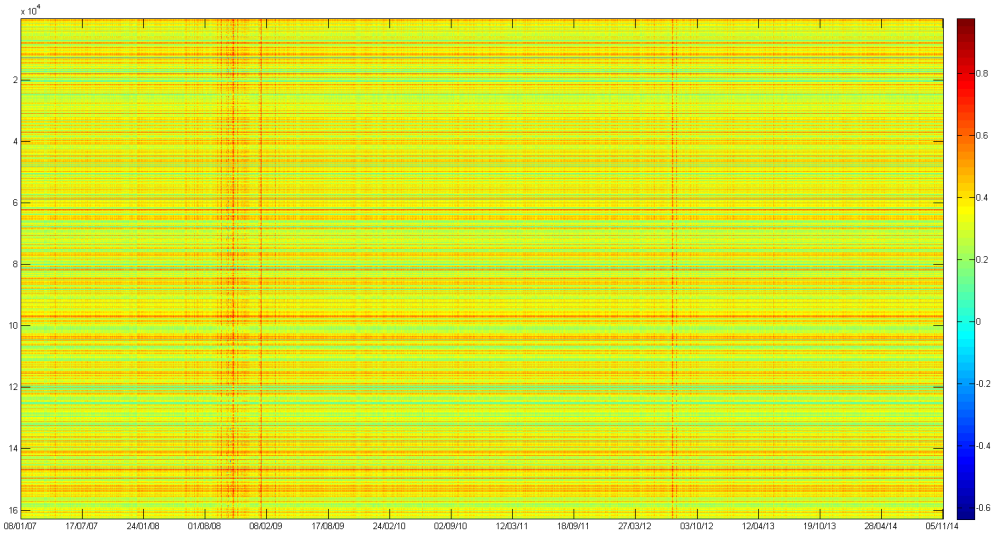

Figure 11: Posterior mean correlations of 571 stocks of *STOXX Europe 600 index*

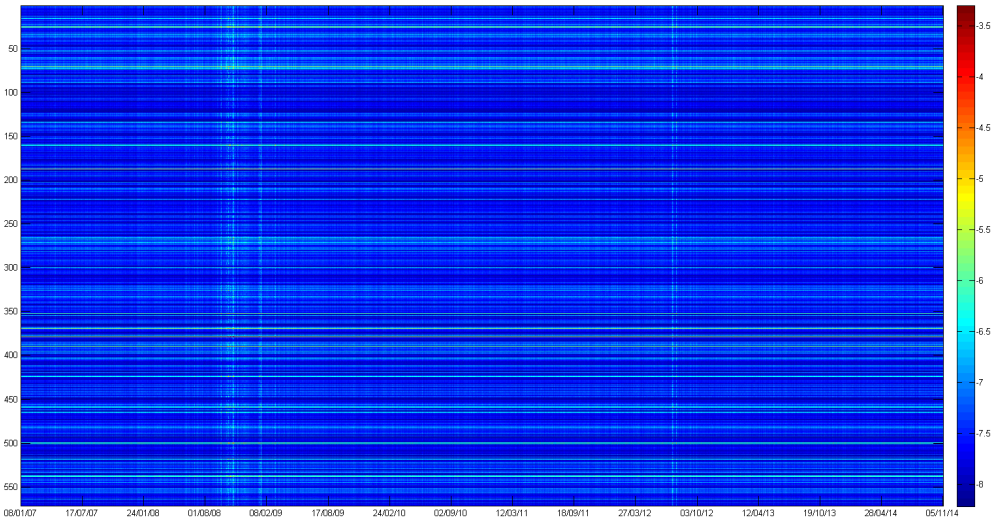

Figure 12: Posterior mean volatilities of 571 stocks of *STOXX Europe 600 index*

7 Conclusion

The contribution of the current paper is to propose a novel multivariate stochastic volatility model which has three important advantages over existing approaches in the literature. First, it allows the entire covariance matrix of potentially large system to evolve stochastically without imposing additional restrictions such as independence across equations or constant off-diagonal elements. Second, our MCMC algorithm is designed to handle large dimensional systems and achieves computational complexity of squared, rather than cubic, order for the evaluation of the likelihood. For very large dimensional problems, we also provide a factor model extension to our MSV model. Third, our multivariate stochastic volatility modelling framework is invariant to permutations of the variables in the system, an issue from which methods using the Cholesky decomposition widely applied macroeconomic VAR models with stochastic volatility suffer from.

The literature on financial econometrics suggests that univariate stochastic volatility models could be enriched by including generalisations such as allowing for non-Gaussian fat-tailed error distributions and/or jumps for the returns and leverage effects expressed through asymmetries in the relation between past negative and positive returns and future volatilities; the review papers by Asai et al. (2006) and Chib et al. (2009) discuss how these can be incorporated in factor models in which the factors are modelled as independent stochastic volatility processes. These are possible important extensions of our approach, but they are not simple in a mutlivariate large-dimensional setup, especially if scalability of the MCMC algorithm is of primary concern, and are left to future research.

Acknowledgement

This work has been supported by the European Union, Seventh Framework Programme FP7/2007-2013 under grant agreement SYRTO-SSH-2012-320270, the The Alan Turing Institute under the Engineering and Physical Sciences Research Council grant EP/N510129/1, the General Directorate for Research in the Government of Catalonia through the Beatriu de Pinós grant 2019/BP/00239 and the Spanish Ministry of Science and Innovation, through the Severo Ochoa Programme for Centres of Excellence in RD (CEX2019-

000915-S). We would like to thank Nicolas Chopin and Omiros Papaspiliopoulos for helpful discussions about Section 5.4.1.

Supplementary material

The Supplementary material provides MCMC specifics and the additional macroeconomic results of Section 6.1, full details about the prior distributions over the parameters $(\theta_h, \theta_\delta)$ and (B, σ^2), a description of the steps for sampling these parameters, and a peudo-code for the recursive algorithm for computing the partial derivatives with respect to the rotation angles.

References

Andersen, T., Bollerslev, T., Lange, S., 1999. Forecasting financial market volatility: Sample frequency vis-à-vis forecast horizon. Journal of Empirical Finance 6, 457–477.

Asai, M., McAleer, M., Yu, J., 2006. Multivariate stochastic volatility: A review. Econometric Reviews 25, 145–175.

Banerjee, S., Gelfand, A.E., Finley, A.O., Sang, H., 2008. Gaussian predictive process models for large spatial data sets. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 70, 825–848.

Barndorff-Nielsen, O., Shephard, N., 2004. Econometric analysis of realised covariation: high frequency based covariance, regression and correlation in financial economics. Econometrica 72, 885–925.

Bauwens, L., Laurent, S., Rombouts, J.V., 2006. Multivariate garch models: a survey. Journal of applied econometrics 21, 79–109.

Bernanke, B., Mihov, I., 1998. Measuring monetary policy. Quarterly Journal of Economics 113, 869–902.

Carpenter, B., Gelman, A., Hoffman, M.D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., Riddell, A., et al., 2017. Stan: A probabilistic programming language. Journal of Statistical Software 76.

Carriero, A., Clark, T., Marcellino, M., 2016. Large vector autoregressions with stochastic volatility and flexible priors. Working Paper .

Carvalho, C.M., Johannes, M.S., Lopes, H.F., Polson, N.G., et al., 2010. Particle learning and smoothing. Statistical Science 25, 88–106.

Carvalho, C.M., West, M., et al., 2007. Dynamic matrix-variate graphical models. Bayesian analysis 2, 69–97.

Chib, S., Omori, Y., Asai, M., 2009. Multivariate stochastic volatility, in: Handbook of Financial Time Series. Springer, pp. 365–400.

Chopin, N., Jacob, P.E., Papaspiliopoulos, O., 2013. Smc2: an efficient algorithm for sequential analysis of state space models. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 75, 397–426.

Chopin, N., Papaspiliopoulos, O., 2020. Introduction to state-space models, in: An Introduction to Sequential Monte Carlo. Springer, pp. 11–25.

Clark, T.E., 2012. Real-time density forecasting from BVARs with stochastic volatility. Journal of Business and Economic Statistics 29, 327–341.

Cogley, T., Primiceri, G.E., Sargent, T.J., 2010. Inflation-gap persistence in the US. American Economic Journal: Macroeconomics 2(1), 43–69.

Cogley, T., Sargent, T.J., 2005. Drifts and volatilities: Monetary policies and outcomes in the post World War II US. Review of Economic Dynamics 8, 262–302.

Cron, A., West, M., 2016. Models of random sparse eigenmatrices and bayesian analysis of multivariate structure, in: Statistical Analysis for High-Dimensional Data. Springer, pp. 125–153.

Daniels, M.J., Kass, R.E., 1999. Nonconjugate bayesian estimation of covariance matrices and its use in hierarchical models. Journal of the American Statistical Association 94, 1254–1263.

Duan, J.C., Fulop, A., 2015. Density-tempered marginalized sequential monte carlo samplers. Journal of Business & Economic Statistics 33, 192–202.

Engle, R., 2002. Dynamic conditional correlation: A simple class of multivariate generalized autoregressive conditional heteroskedasticity models. Journal of Business and Economics Statistics 20, 339–350.

Forni, M., Hallin, M., Lippi, M., Reichlin, L., 2000. The generalized dynamic-factor model: Identification and estimation. Review of Economics and statistics 82, 540–554.

Fulop, A., Li, J., 2013. Efficient learning via simulation: A marginalized resample-move approach. Journal of Econometrics 176, 146–161.

Fulop, A., Li, J., 2019. Bayesian estimation of dynamic asset pricing models with informative observations. Journal of Econometrics 209, 114–138.

Gerber, M., Chopin, N., 2015. Sequential quasi monte carlo. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 77, 509–579.

Geweke, J., Amisano, G., 2010. Comparing and evaluating bayesian predictive distributions of asset returns. International Journal of Forecasting 26, 216–230.

Girolami, M., Calderhead, B., 2011. Riemann manifold langevin and hamiltonian monte carlo methods. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 73, 123–214.

Gouriéroux, C., Jasiak, J., Sufana, R., 2009. The wishart autoregressive process of multivariate stochastic volatility. Journal of Econometrics 150, 167–181.

Harvey, A., Ruiz, E., Shephard, N., 1994. Multivariate stochastic variance models. The Review of Economic Studies 61, 247–264.

Jacquier, E., Polson, N.G., Rossi, P., 1994a. Bayesian analysis of stochastic volatility models. Journal of Business and Economic Statistics 12, 371–418.

Jacquier, E., Polson, N.G., Rossi, P.E., 1994b. Bayesian analysis of stochastic volatility models. Journal of Business & Economic Statistics 12, 371–89.

Kass, R.E., Raftery, A.E., 1995. Bayes factors. Journal of the american statistical association 90, 773–795.

Kim, C., Nelson, C.R., 1999. Has the U.S. Economy become more stable? A Bayesian Approach based on a Markov-Switching Model of the Business Cycle. Review of Economics and Statistics 81, 608–618.

Kim, S., Shephard, N., Chib, S., 1998a. Stochastic volatility: likelihood inference and comparison with arch models. The Review of Economic Studies 65, 361–393.

Kim, S., Shephard, N., Chib, S., 1998b. Stochastic volatility: likelihood inference and comparison with ARCH models. Review of Economic Studies 65, 361–393.

Koop, G., Korobilis, D., 2013. Large time-varying parameter VARs. Journal of Econometrics 177, 185–198.

Koop, G., Korobilis, D., 2019. Variational Bayes inference in high-dimensional time-varying parameter models. Working paper .

Lewandowski, D., Kurowicka, D., Joe, H., 2009. Generating random correlation matrices based on vines and extended onion method. Journal of multivariate analysis 100, 1989–2001.

McConnell, M., Perez Quiros, G., 2000. Output fluctuations in the U.S.: what has changed since the early 1980s? American Economic Review 90, 1464–1476.

Petrova, K., 2019. A quasi-Bayesian local likelihood approach to time varying parameter VAR models. Journal of Econometrics 212, 286–306.

Philipov, A., Glickman, M.E., 2006a. Factor multivariate stochastic volatility via wishart processes. Econometric Reviews 25, 311–334.

Philipov, A., Glickman, M.E., 2006b. Multivariate stochastic volatility via wishart processes. Journal of Business & Economic Statistics 24, 313–328.

Pitt, M., Shephard, N., 1999a. Time varying covariances: a factor stochastic volatility approach. Bayesian statistics 6, 547–570.

Pitt, M.K., Shephard, N., 1999b. Filtering via simulation: Auxiliary particle filters. Journal of the American statistical association 94, 590–599.

Platanioti, K., McCoy, E., Stephens, D., 2005. A review of stochastic volatility: univariate and multivariate models. Technical Report. Imperial College London.

Primiceri, G., 2005. Time-varying structural vector autoregressions and monetary policy. Review of Economic Studies 72, 821–852.

Ram, N., Brose, A., Molenaar, P.C., 2013. Dynamic factor analysis: Modeling person-specific process. The Oxford Handbook of Quantitative Methods in Psychology: Vol. 2: Statistical Analysis 2, 441.

Rebeschini, P., Van Handel, R., 2015. Can local particle filters beat the curse of dimensionality? The Annals of Applied Probability 25, 2809–2866.

Sims, C.A., 1980. Macroeconomics and reality. Econometrica 48, 1–48.

Sims, C.A., Zha, T., 2006. Were there regime switches in U.S. monetary policy? American Economic Review 96, 1193–1224.

Taylor, S., 1986. Modelling Financial Time Series. John Wiley, Chichester.

Tims, B., Mahieu, R., 2003. A range-based multivariate model for exchange rate volatility .

Titsias, M., 2011. Contribution to the discussion of the paper by girolami and calderhead. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 73, 197–199.

Titsias, M.K., Papaspiliopoulos, O., 2018. Auxiliary gradient-based sampling algorithms. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 80, 749–767.

Tsay, R., 2005. Analysis of Financial Time Series. John Wiley, New York.

Wilson, A., Ghahramani, Z., 2010. Generalised wishart processes. Uncertainty in Artificial Intelligence (2011) .

Yang, R., Berger, J.O., 1994. Estimation of a covariance matrix using the reference prior. The Annals of Statistics , 1195–1211.