We modeled long memory with just one lag!

Luc Bauwens, Guillaume Chevillon, Sébastien Laurent

We modeled long memory with just one lag!*

Luc Bauwens[a], Guillaume Chevillon[b], and Sébastien Laurent[c]

[a]LIDAM/CORE, UCLouvain, Voie du Roman Pays 34, 1348 Louvain-La-Neuve, Belgium;

[b]ESSEC Business School;

[c]Aix-Marseille University (Aix-Marseille School of Economics), CNRS, EHESS,
Aix-Marseille Graduate School of Management – IAE

April 25, 2023

Abstract

Two recent contributions have found conditions for large dimensional networks or systems to generate long memory in their individual components. We build on these and provide a multivariate methodology for modeling and forecasting series displaying long range dependence. We model long memory properties within a vector autoregressive system of order 1 and consider Bayesian estimation or ridge regression. For these, we derive a theory-driven parametric setting that informs a prior distribution or a shrinkage target. Our proposal significantly outperforms univariate time series long-memory models when forecasting a daily volatility measure for 250 U.S. company stocks over twelve years. This provides an empirical validation of the theoretical results showing long memory can be sourced to marginalization within a large dimensional system.

Keywords: Bayesian estimation, Ridge regression, Vector autoregressive model, Forecasting.
JEL: C10, C32, C58.

*Guillaume Chevillon acknowledges financial support from Institut Europlace de Finance & Labex Louis Bachelier, ESSEC Foundation and CERESSEC, and Labex MME-DII. Sébastien Laurent acknowledges the research support of the French National Research Agency Grants ANR-17-EURE-0020 and ANR-21-CE26-0007-01 and the Excellence Initiative of Aix-Marseille University - A*MIDEX. The authors are grateful for the helpful discussions with seminar participants at Aix-Marseille, the Bank of Japan, the Joint Research Centers of the European Commission at Ispra, Lancaster, Lund, Maastricht, Macquarie, Nottingham, Oxford, Paris School of Economics, Queen Mary University, SMU, Sydney University, the University of New South Wales, as well as the QFFE 2022 conference in Marseille, the 2019 EC^2 in Oxford, the 4th Vienna workshop on Time Series, the first Aarhus Econometrics Workshop, and the workshops on Financial Econometrics at Orebro University and on Long Memory in Hannover, for helpful discussions. We thank in particular Christian Brownlees, Matei Demetrescu, Jurgen Doornik, Domenico Giannone, Uwe Hassler, Alain Hecq, Liudas Giraitis, Roselyne Joyeux, Sebastiano Manzan, Sophocles Mavroeidis, Ulrich Müller, Bent Nielsen, Morten Nielsen, Susanne Schennach, Toshitaka Sekine, Shuping Shi, Philipp Sibbertsen, Lorenzo Trapani and Jun Yu for useful comments. We are in particular grateful to two anonymous referees, one Associate Editor and the Managing Editor, Serena Ng, for their careful readings and insightful comments that greatly improved the quality of the paper.

1 Introduction

Long memory, i.e., a dependence between observations decaying hyperbolically with their distance in time (see, e.g., Beran, 1992), is often encountered in economic and financial time series, and long memory models have been found to provide a good empirical representation of persistence that is stronger than stationary ARMA but weaker than unit-root processes.

The literature has found several potential sources of long memory, and our interest in this paper concerns specifically two contributions that show that long memory can arise in individual series that are linked within an infinite dimensional network or system. Chevillon, Hecq and Laurent (2018, CHL henceforth) prove that long memory can result from the marginalization of a large dimensional system. More specifically, they provide a parametric framework in which the variables of an n-dimensional vector autoregressive model of order one, i.e., a VAR(1), can be individually modelled as independent fractionally integrated white noises (defined as in Granger and Joyeux, 1980) as n tends to infinity. Long memory may therefore be a feature of univariate or low dimensional models that vanishes when considering larger systems in their entirety: while the infinite dimensional system is Markovian, modeling the series individually requires infinite lags. Working on dynamics of networks of infinite dimension, Schennach (2018, Schennach henceforth) has found a related result of hyperbolic response of outputs to distant input shocks. In linear networks, her results can also specialize to VAR(1) dynamics.

Motivated by the theoretical result that a large VAR system of finite order can asymptotically (in the number of variables) generate long memory in its individual components, our main contribution is to show that this result is empirically relevant, and to provide operational techniques for implementing it fruitfully in applied work. Our strategy relies on designing a *long memory prone* target model that combines the key characteristics of CHL and Schennach: it specifies, in the case of a VAR(1), the magnitude of the on- and off-diagonal elements and their sums. Since we operate in a context where there are many parameters to estimate, relative to the number of observations, we design an empirical procedure that uses our long memory prone target via shrinkage (L2 penalization) or a Bayesian prior. We recommend equation-by-equation estimation where each equation consists of an AR(1) (autoregressive of order 1) model augmented by the first lag of all the other variables in the system. The degree of shrinkage, which is governed by the L2 penalty weights or by the prior variances, is chosen by cross-validation between the two extremes of dogmatic, or no, restrictions. We illustrate the inferential performance of the strategy by simulations, where we show we can uncover whether the source of long memory resides in the dependence in the system or time dimensions. Given the large dimensions we operate with, we rely for model evaluation on an out-of-sample forecasting competition against standard univariate long memory models (namely, the ARFIMA model, and the HAR model of Corsi, 2009). We also assess in-sample whether long memory is captured through measurements of residual serial dependence.

We perform an empirical application in a context where long memory and multivariate interdependence are established features, and where data with long time series and large cross-sectional dimensions can be obtained. We therefore focus on the logarithm of a robust-

to-jumps estimate of the daily integrated variance computed from 5-minute returns for 250 US stocks over twelve years. Since we work with reduced form forecasting models based on different information sets, it is sensible to use measures of forecast accuracy as comparison criteria. Hence, we compare the forecasts, at horizons ranging from $h = 1$ to 50, produced by the different models. We use the mean squared forecast error loss function, and we rely on the model confidence set procedure of Hansen, Lunde and Nason (2011) to discriminate between the models. We find that our proposal for shrinkage towards, or a prior centered on, a multivariate long memory prone target successfully outperforms univariate models of long memory at all horizons (depending on the horizon, we achieve between 5% and 20% reduction in average mean squared forecast error compared to ARFIMA and HAR models). This indicates that, in our dataset, long memory observed in individual series may likely be caused by spillovers within the system.

The paper is organized as follows. Section 2 introduces the theoretical framework under which a VAR(1) model can generate long memory in its components when the dimension of the system is large. The theory implies restrictions on the VAR parameters that we collect in a target model that is related to the literature on dynamic factor models. Section 3 shows how to use these restrictions to estimate the parameters of the VAR(1) model, either through an informative prior density for conducting Bayesian estimation, or by ridge estimation involving nonzero targets. Section 4 compares the in-sample fit and forecasting performance of the proposed and competing methods via Monte Carlo simulations. Section 5 presents the empirical analysis and Section 6 our conclusions. Proofs, technical details and additional figures are collected in an appendix. Throughout, we use the notation $a_n \asymp b_n$ to denote equal magnitude as $n \to \infty$, i.e., both $a_n = O(b_n)$ and $b_n = O(a_n)$, and similarly for $\asymp_p$ and $O_p(\cdot)$.

2 Long memory in a VAR(1) system

This section reviews the elements of the theoretical frameworks of CHL and Schennach that preside over our own modeling strategy. We provide a unifying treatment and derive constraints that are germane to our estimation procedures.

Both CHL and Schennach prove that long memory observed in a univariate time series can be the result of the marginalization of an infinitely large VAR(1) system that satisfies some specific assumptions. For this reason, we let the observable vector $\boldsymbol{y}_{n,t}$ of dimension n satisfy, for $t \geq 1$,

$$(\boldsymbol{I}_n - \boldsymbol{A}_n L)(\boldsymbol{y}_{n,t} - \boldsymbol{\mu}) = \boldsymbol{\epsilon}_{n,t}, \tag{1}$$

where $\boldsymbol{\epsilon}_{n,t}$ is a short memory process with zero expectation and variance-covariance matrix $\boldsymbol{\Sigma}_n$ with bounded L1 norm.

In order to reduce expositional complexity and simplify their derivations, both CHL and Schennach restrict themselves to matrices that belong to the Toeplitz family since these require only $O(n)$ parameters. While their high-level assumptions differ, all can be subsumed in

$$\boldsymbol{A}_n = \boldsymbol{T}_n + \eta_n \boldsymbol{D}_n,$$

where η_n is a vanishing scalar sequence, and $\{\boldsymbol{T}_n\}$ and $\{\boldsymbol{D}_n\}$ denote generic sequences of Toeplitz matrices that are, respectively, symmetric and antisymmetric. Both CHL and Schennach assume that $\{\boldsymbol{D}_n\}$ plays no role asymptotically, so large-system dynamics are governed by $\boldsymbol{T}_n$, the entries of which are labelled as

$$\boldsymbol{T}_n = \begin{bmatrix} t_0^{(n)} & t_1^{(n)} & \cdots & t_{n-1}^{(n)} \\ t_1^{(n)} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & t_1^{(n)} \\ t_{n-1}^{(n)} & \cdots & t_1^{(n)} & t_0^{(n)} \end{bmatrix}. \tag{2}$$

In Schennach, the process $\boldsymbol{y}_{n,t}$ can be seen as generated by a network that lies in a space of dimension one (a linear network). She also considers higher dimensions (hence the Toeplitz assumption to control complexity), but for the purpose of the analysis, we restrict ourselves to a one-dimensional linear network, and assume each node lies in $\mathbb{N}$. In the spirit of Diebold and Yılmaz (2009, 2014), who model connectedness within a network using a VAR model, this amounts to a system that consists of an infinite but countable number of variables indexed by $j \in \mathbb{N}$. We denote the limiting, infinite dimensional, vectors by $(\boldsymbol{y}_t, \boldsymbol{\epsilon}_t) = \lim_{n\to\infty} (\boldsymbol{y}_{n,t}, \boldsymbol{\epsilon}_{n,t})$, and the ith elements of $\boldsymbol{y}_t, \boldsymbol{\epsilon}_t$ by $y_t^{(j)}, \epsilon_t^{(j)}$, for $j \in \mathbb{N}$. We next describe the two models that have been shown to generate long memory within an infinite dimensional VAR(1) model such as (1).

Chevillon et al. (2018, CHL). These authors make a set of parametric assumptions (their Assumption T) where they specify a mapping such that entries of $\boldsymbol{T}_n$ only depend on a scalar sequence $\delta_n \in (0, 1/2)$ satisfying $n^2(\delta_n - 1/2) = o(1)$. Their Assumption T implies in particular that, as $n \to \infty$, with $(n-1)/4 \in \mathbb{N}$,

$$t_0^{(n)} \to 1/2, \tag{3a}$$

$$t_k^{(n)} = O\left(n^{-1}\right), \quad \text{for } k \neq 0, \tag{3b}$$

$$\sum\nolimits_{k=0}^{n-1} t_k^{(n)} = 1. \tag{3c}$$

Under the additional assumption $\boldsymbol{\epsilon}_{n,t} \sim \mathsf{NID}(\mathbf{0}, \boldsymbol{\Sigma}_n)$, with $\boldsymbol{\Sigma}_n$ diagonal, they prove (in their Theorem 1) that, as $n \to \infty$, all components of $\boldsymbol{y}_{n,t}$ tend to independent fractional white noises with identical degrees of integration (all equal to 1/2), so for all $j \in \mathbb{N}$:

$$y_{n,t}^{(j)} \Rightarrow \mu^{(j)} + \Delta^{-1/2}\epsilon_t^{(j)},$$

where $\Delta = 1 - L$ and $\Rightarrow$ denotes weak convergence of the associated probability measures. Since the entries of $\boldsymbol{A}_n - \frac{1}{2}\boldsymbol{I}_n$ tend to zero as $n \to \infty$, the cross-sectional dependence between the elements of $\boldsymbol{y}_{n,t}$ vanishes as $n \to \infty$. Yet, as in this setting $\sum_{k=0}^{n-1} t_k^{(n)} = 1$ remains nonzero, the dependence across individual series is sufficient to generate long memory in each of the components of the multivariate process.

Schennach (2018). She considers the limiting structure where $\boldsymbol{T} = \lim_{n\to\infty} \boldsymbol{A}_n = \lim_{n\to\infty} \boldsymbol{T}_n$, i.e., the case of an infinite dimensional network. She assumes that $\boldsymbol{\epsilon}_t$ constitutes a short memory MA(∞) process. The entries (t_k) of $\boldsymbol{T}$ are assumed to be nonnegative and satisfy

$$t_0 > 0, \tag{4a}$$

$$\mathsf{card}\left\{k \in \mathbb{N},\ t_k > 0\right\} < \infty, \tag{4b}$$

$$\sum\nolimits_{k=0}^{\infty} t_k = 1. \tag{4c}$$

She proves (in her Theorem 4) that, for all i, j, there exists a $c_{ij} > 0$ such that, as $k \to \infty$,

$$\frac{\partial y_{t+k}^{(i)}}{\partial \epsilon_t^{(j)}} = c_{ij} k^{-1/2} + O\left(k^{-3/2}\right),$$

i.e., the impulse response function of $y_{t+k}^{(i)}$ to a shock $\epsilon_t^{(j)}$ is hyperbolic and its speed of decay corresponds to that of a process that is integrated of order 1/2.

Both Schennach and CHLChevillon, Hecq, and Laurent (2018) find long memory of fractional degree one-half in infinite dimensional networks. They use different approaches and assumptions, but rely on the Toeplitz nature of dependence across the infinite – yet countable – number of variables in the system (or nodes in the network). Both of them consider matrices whose rows and columns sum to unity. Schennach focuses on the interactions within the limiting system while CHL consider the evolution in dynamics as the finite system grows larger.

Schennach's assumptions on $\boldsymbol{\epsilon}_t$ are less restrictive. She also does not specify the values of the entries of $\boldsymbol{A}$ but assumes that only a finite number of t_k coefficients are nonzero, so that a rotation of $\boldsymbol{A}$ is *banded* (i.e., all subdiagonals are zero beyond a point). Hence, the system she considers (i.e., the one dimensional version) is *sparse*, each variable being only directly connected to a finite number of variables. By contrast, CHL rely on *i.i.d.* shocks and make parametric assumptions on $\boldsymbol{T}_n$. Their setting is *dense*, variables being directly connected to all other variables, but with a connection that becomes weaker as the dimension of the system increases.

Schennach's result is, then, that all response functions of all variables to all shocks exhibit hyperbolic decay, whereas CHL's applies only to the responses of variables to their idiosyncratic shocks in the VAR system.

Target proposal. The similarities between Equations (3a)-(3c) and (4a)-(4c) are clear. The main differences relate to specifications of the Toeplitz assumptions, (3b)-(4b) in particular. In empirical work, the Toeplitz assumption *unreasonably* requires a specific ordering of the variables so we cannot retain it. This implies that we cannot either assume without extra knowledge that specific variables are unconnected. Hence, denoting by $a_{ij}^{(n)}$ the entries of $\boldsymbol{A}_n$ in Equation (1), the model can be said to be *long memory prone*, i.e., compatible with the theoretical results of both CHL and Schennach, if there exist 'small' $\varepsilon, \varepsilon' > 0$ such that for all (i, j),

C1: the autoregressive coefficients $\left(a_{ii}^{(n)}\right)$ are close to 1/2:

$$a_{ii}^{(n)} \in (1/2 - \varepsilon, 1/2],$$

C2: the off-diagonal elements $\left(a_{ij}^{(n)}\right)_{i \neq j}$ are nonnegative, close to 0 and of order $O(n^{-1})$:

$$0 \leq n a_{ij}^{(n)} < \varepsilon',$$

C3: the sums of row and column elements of $\boldsymbol{A}_n$ are equal to 1:

$$\sum\nolimits_{k=0}^{\infty} a_{ik}^{(n)} = \sum\nolimits_{k=0}^{\infty} a_{kj}^{(n)} = 1.$$

A simple family of matrices that aligns with Conditions **C1**-**C3** consists in setting, for some d_0 close to 1/2,

$$\boldsymbol{A}_n^0 = d_0 \boldsymbol{I}_n + \frac{1 - d_0}{n - 1} (\boldsymbol{J}_n - \boldsymbol{I}_n), \tag{5}$$

where $\boldsymbol{J}_n$ is a square matrix of dimension n whose elements are all equal to unity. Hence, for all i, j, the entries of $\boldsymbol{A}_n^0$ satisfy $a_{ii}^{(n)} = d_0$ and $a_{ij}^{(n)} = (n-1)^{-1}(1 - d_0)$ if $i \neq j$. In practice, we do not expect $\boldsymbol{A}_n^0$ to hold exactly but it constitutes a proposal for a *long memory prone* target of a shrinkage estimator or, in a Bayesian context, the mean of a prior distribution.

Conditions **C1**-**C3** participate to the empirical methodology we propose in the next sections but we, first, explore the link of target $\boldsymbol{A}_n^0$ with factor models.

Factor representation. Under the proposed target $\boldsymbol{A}_n^0$, off-diagonal elements are an order of magnitude smaller than those on the diagonal, yet each row sums to unity. Hence, each individual equation in the VAR, $y_{i,t} = a_{ii}^{(n)} y_{i,t-1} + \sum_{j=1, j \neq i}^{n} a_{ij}^{(n)} y_{j,t-1} + \epsilon_{i,t}$ writes as a weighted average of the idiosyncratic innovation $\epsilon_{i,t}$, the lagged variable $y_{i,t-1}$, and of a common factor – the cross-sectional average $\bar{y}_t = n^{-1} \sum_{j=1}^{n} y_{j,t-1}$ – all with non-degenerate weights:

$$y_{i,t} = \frac{n d_0 - 1}{n - 1} y_{i,t-1} + (1 - d_0) \frac{n}{n-1} \bar{y}_{t-1} + \epsilon_{i,t}. \tag{6}$$

Yet, denoting by $\mathbf{i}_n = (1, ..., 1)'$ the n-vector of ones, and under condition **C3** that columns of $\boldsymbol{A}_n$ sum to unity, we see that $\mathbf{i}_n' (\mathbf{I}_n - \boldsymbol{A}_n) = 0$ and $\mathbf{i}_n' (\boldsymbol{I}_n - \boldsymbol{A}_n L) = (1 - L) \mathbf{i}_n'$. Hence, for finite n, the cross-sectional average process $\bar{y}_t = n^{-1} \mathbf{i}_n' \boldsymbol{y}_{n,t}$ follows a random walk:

$$\bar{y}_t = \bar{y}_{t-1} + \bar{\epsilon}_t,$$

where $\bar{\epsilon}_t = n^{-1} \sum_{j=1}^{n} \epsilon_{j,t}$, and the variance of this average innovation decreases when n gets large, i.e., $\mathsf{Var}(\bar{\epsilon}_t) = n^{-2} \mathbf{i}_n' \boldsymbol{\Sigma}_n \mathbf{i}_n \asymp n^{-1}$ since $\boldsymbol{\Sigma}_n$ with entries $\Sigma_n^{(ij)}$ has bounded L1 norm, defined by $\|\boldsymbol{\Sigma}_n\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^{n} \left|\Sigma_n^{(ij)}\right|$. The average $\bar{y}_t$ therefore corresponds to a *damped*

stochastic trend process: $\bar{y}_t = O_p\left(\sqrt{T/n}\right) = o_p(1)$, where the sample size T is assumed to be fixed, and we assume that $\bar{y}_0 \asymp_p 1$. The attenuation of the random walk is due to cross-sectional averaging. This differs from the temporal averaging considered in the long-range dependence literature, such as, e.g., the "local level" model $y_t = \frac{1}{T}\sum_{s=1}^{t}\eta_s + \epsilon_t$ of Harvey (1990) that Müller and Watson (2008) study as a model for long range persistence constituting an alternative to fractional integration and near stochastic trends.

Each equation of the system therefore involves one autoregressive element and the *cross-sectional average* common factor (see Pesaran, 2006) that follows a damped stochastic trend. Hence, although the data generated by (6) exhibits a stochastically trending common factor that impacts all variables, its damped nature for finite T (the asymptotic degeneracy of $\mathsf{Var}(\bar{y}_{t-1})$ as n gets large) implies that $\mathbf{y}_{n,t}$ is cross-sectionally *weakly* dependent following the definition of Chudik and Pesaran (2011) – note, however, that our setting violates their Assumption 4 of stationarity.

In practice, we do not expect the target to hold exactly, but the discussion above shows that the multivariate source of long memory is close to a model admitting an equivalent factor representation (possibly via a cross-sectional weighted average of the variables). The latter's implied dynamics bear some resemblance with the local-level (damped trend) model studied in the literature, by Müller and Watson (2008) in particular, as an alternative source of long range persistence, with the difference that, here, the stochastic trend is a cross-sectional rather than a time series average.

3 Methodology for long memory prone estimation

We turn to the question of estimating $\boldsymbol{A}_n$ when $\boldsymbol{y}_{n,t}$ may exhibit long range dependence. We present a methodology that shrinks the estimates of $\boldsymbol{A}_n$ in a manner that is informed by the stylized conditions **C1-C3** combining the frameworks of CHL and Schennach. Indeed, it does not seem efficient when the system has a large dimension to ignore these stylized assumptions altogether, and estimate the VAR by ordinary least-squares (OLS).

An obvious approach to being informed by **C1-C3** consists in imposing them strictly such as, e.g., via parametrizing explicitly the elements of $\boldsymbol{A}_n$. For instance CHL use in their Assumption T a mapping that defines all the elements of $\boldsymbol{A}_n$ through a single scalar δ_n. The latter could be estimated by minimum distance or maximum likelihood (ML). This is certainly too restrictive as explained in the previous section, and we may want to retain a certain degree of flexibility around these restrictions.

We therefore consider intermediate strategies. One of them is penalized regression (e.g., ridge or the LASSO), where the least squares criterion is augmented by restrictions whose strength is modulated through penalty parameters. The resulting estimator is shrunk in the direction of the restrictions. We prefer to use ridge estimation rather than the LASSO that leads to estimating some of the coefficients *exactly* at their target values. Indeed, since our target model differs from CHL and Schennach, we do not want to impose it exactly (as would arise with L1 regularization around the target), nor do we want to exclude specific variables (as in the standard LASSO). Since L2 regularization does not impose the target

values, ridge estimation permits more ambivalence when it comes to estimating parameters close to our target model.

Bayesian estimation provides another intermediate method, whereby the restrictions are embedded in a prior density, so that they hold a priori on average (through the prior expectation of the parameters), but with some degree of uncertainty (through prior positive variances on the parameters or functions thereof). Depending on the degree of tightness of the prior, the prior information pulls the data information more or less strongly in the direction of the restrictions. Several authors have contributed to Bayesian estimation of VAR models, using different types of prior information, see Karlsson (2013) for a review. The types of restrictions considered in the literature, such as the so-called "Minnesota prior" for unit roots (see Doan, Litterman, and Sims, 1984), or the "long run" forecasting restrictions (Giannone, Lenza, and Primiceri, 2019) are relevant to modeling and forecasting short memory macroeconomic time series. Our contribution differs in that we use a prior density that shrinks the parameters to values informed by long memory prone restrictions.

We detail the regression model in the next subsection, and suggest an "equation-by-equation" estimation of the VAR system. Our approaches to ridge regression and Bayesian estimation are exposed in Subsections 3.2 and 3.3, respectively. We denote the resulting models estimated by ridge or Bayesian methods by RAR-X and BAR-X.

3.1 Framework

We consider the estimation of a VAR(1) system, written at date t (dropping the subscript n on $\boldsymbol{A}_n$ and on the processes) as

$$\boldsymbol{y}_t = \boldsymbol{\tau} + \boldsymbol{A}\boldsymbol{y}_{t-1} + \boldsymbol{\epsilon_t}, \tag{7}$$

for the vector $\boldsymbol{y}_t$ consisting of n variables. In this paper, we suggest to estimate parameters $\boldsymbol{\tau}$ and $\boldsymbol{A}$ "equation-by-equation", instead of globally for the entire system. Assuming $\boldsymbol{\epsilon_t}$ is multivariate Gaussian with zero expectation and constant covariance matrix $\boldsymbol{\Sigma}$, then estimating each equation separately by OLS is equivalent to estimating the system jointly by Maximum Likelihood (ML), even if $\boldsymbol{\Sigma}$ is not diagonal. For Bayesian estimation, equation-by-equation estimation is not equivalent to the joint estimation of all equations, but the latter approach is much more demanding in computing time for the dimensions we are interested in (e.g., 250 as in the empirical application).

A typical equation of the VAR(1) system is an AR-X – dropping (1)– regression equation that is written at date t as

$$y_t = \gamma_0 + \boldsymbol{\gamma}'\boldsymbol{x}_t + \epsilon_t, \tag{8}$$

where y_t denotes a variable of the system, γ_0 is the intercept parameter, $\boldsymbol{x}_t$ is the column vector containing the first lag of the n variables of the system (*including* the first lag of y_t), $\boldsymbol{\gamma} = (\gamma_1, \gamma_2, \ldots, \gamma_n)'$ is the vector of n slope coefficients of $\boldsymbol{x}_t$, and ϵ_t is an error term assumed to be Gaussian with zero expectation and constant variance σ^2. By convention, for any variable of the VAR, $\boldsymbol{x}_t$ is ordered in such a way that its first element is the lagged dependent variable (y_{t-1}), and $\boldsymbol{\gamma}$ is ordered accordingly: its first element (γ_1) is the autoregressive

coefficient of the dependent variable, and the remaining elements are the coefficients of the other lagged variables. For example, if y_t is the first element of $\boldsymbol{y}_t$, $\boldsymbol{\gamma}'$ is the first row of matrix $\boldsymbol{A}$, and γ_0 is the first element of $\boldsymbol{\tau}$.

Over a sample of T observations, write the AR-X equation in the standard regression notation

$$\boldsymbol{Y} = \boldsymbol{Z}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{9}$$

where $\boldsymbol{Y} = (y_1, y_2, \ldots, y_T)'$, $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_T)' \sim \mathsf{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{I_T})$, $\boldsymbol{Z}$ is a $T \times k$ matrix, with $k = 1 + n$ and t-th row equal to $(1, \boldsymbol{x}_t')$, and $\boldsymbol{\beta} = (\gamma_0, \boldsymbol{\gamma}')'$.

Estimation of $\boldsymbol{\beta}$ by OLS is likely to be imprecise when n is large compared to T, and this will affect the quality of forecasts negatively. To align with the stylized conditions **C1-C3**, we recommend shrinking the elements of the vector $\boldsymbol{\beta} = (\gamma_0, \boldsymbol{\gamma}')'$ in (9) to a target such as $\boldsymbol{A}_n^0$ defined in Equation (5). Next, we suggest ways of introducing these conditions through ridge and Bayesian estimation.

3.2 Ridge estimation

To achieve conditions **C1-C3**, we define as the shrinkage target of $\boldsymbol{\beta}$ the vector

$$\boldsymbol{\beta}_0 = (0, d_0, a_0, \ldots, a_0)', \tag{10}$$

where $a_0 = (1 - d_0)/(n - 1)$ is repeated $n - 1$ times, as in the first row of $\boldsymbol{A}_n^0$ defined by (5). The scalar $d_0 \in (0, 1)$ is the target for the autoregressive coefficient and it determines the target a_0 of the other coefficients that are shrunk to a value that is close to zero when n is large. We allow $d_0 > 1/2$ despite condition **C1** to avoid boundary effects. We use two penalty parameters to control the shrinkage strength: λ_d^2 for the autoregressive parameter, and λ_a^2 for the other coefficients. The penalty function is defined as

$$\lambda_d^2(\gamma_1 - d_0)^2 + \lambda_a^2 \sum_{i=2}^{n} (\gamma_i - a_0)^2 = (\boldsymbol{\beta} - \boldsymbol{\beta}_0)' \boldsymbol{\Lambda}_k (\boldsymbol{\beta} - \boldsymbol{\beta}_0), \tag{11}$$

where $\boldsymbol{\Lambda}_k = \mathrm{diag}(0, \lambda_\mathrm{d}^2, \lambda_\mathrm{a}^2, \ldots, \lambda_\mathrm{a}^2)$.

In this way, the last n elements of $\boldsymbol{\beta}$ are shrunk to the corresponding elements of $\boldsymbol{\beta}_0$, but the first element of $\boldsymbol{\beta}$ is not shrunk, the value (zero) of the first element of $\boldsymbol{\beta}_0$ being practically irrelevant.

The choice of $\boldsymbol{\beta}_0$ implies that the sum of the last n coefficients is equal to 1 in the target, but the penalty is distributed over the n coefficients. To better achieve **C3**, we add the penalty term $\lambda_s^2(\boldsymbol{\iota}'\boldsymbol{\beta} - \boldsymbol{\iota}'\boldsymbol{\beta}_0)^2$, where λ_s^2 is a penalty parameter and $\boldsymbol{\iota} = (0, 1, 1, \ldots, 1)'$ is a vector of k elements.

The extended ridge (ER) estimator is obtained by minimizing the objective function

$$(\boldsymbol{Y} - \boldsymbol{Z}\boldsymbol{\beta})'(\boldsymbol{Y} - \boldsymbol{Z}\boldsymbol{\beta}) + (\boldsymbol{\beta} - \boldsymbol{\beta}_0)' \boldsymbol{\Lambda}_k (\boldsymbol{\beta} - \boldsymbol{\beta}_0) + \lambda_s^2(\boldsymbol{\iota}'\boldsymbol{\beta} - \boldsymbol{\iota}'\boldsymbol{\beta}_0)^2, \tag{12}$$

and can be shown to be (see Appendix A)

$$\boldsymbol{\beta}_{ER} = \left(\boldsymbol{Z}'\boldsymbol{Z} + \boldsymbol{\Lambda}_k + \lambda_s^2 \boldsymbol{\iota}\boldsymbol{\iota}'\right)^{-1} (\boldsymbol{Z}'\boldsymbol{Y} + \boldsymbol{\Lambda}_k \boldsymbol{\beta}_0 + \lambda_s^2 \boldsymbol{\iota}\boldsymbol{\iota}'\boldsymbol{\beta}_0). \tag{13}$$

As usual, the ridge estimator simplifies to the OLS estimator if all the penalty parameters are set to zero.

The values of d_0, λ_d^2, λ_a^2, and λ_s^2 can be chosen by cross-validation on a training sample. A grid of values is set a priori for each of them. For each point of the grid, the estimator is computed using 80 percent of the training sample, forecasts are computed for the last 20 percent, and a forecast loss function is computed. The chosen quadruplet is the value minimizing the loss function over the grid. After this step, estimation is performed on a subsequent sample, and forecasts are computed and evaluated over a post-estimation sample. Details are provided in Section 4.

3.3 Bayesian estimation

Bayesian estimation is based on a prior density for $\boldsymbol{\beta}$ and σ^2, and the likelihood function, the latter resulting from the assumption of normality of the error terms. Since the theory does not provide information on σ^2, its prior "density" $p(\sigma^2)$ is chosen to be the usual "non-informative" prior:

$$p(\sigma^2) \propto 1/\sigma^2. \tag{14}$$

The prior density of $\boldsymbol{\beta}$ is designed to include the theory restrictions, conditions **C1-C3**. We opt for a Gaussian density for three reasons: (i) it is convenient for computing the posterior density (see Section B of the appendix); (ii) implementation of the restrictions is easily done using four scalar parameters, as explained below; and (iii) the restrictions do not explicitly require an asymmetric density. The prior density is proportional to

$$\exp[-\frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\beta}_0)'\boldsymbol{Q}_0(\boldsymbol{\beta}-\boldsymbol{\beta}_0)]\exp[-\frac{1}{2}h_0(\boldsymbol{\beta}'\boldsymbol{\iota}-\boldsymbol{\beta}_0'\boldsymbol{\iota})^2]. \tag{15}$$

The vector $\boldsymbol{\beta}_0$ is defined as in (10) and depends on the scalar hyperparameter d_0 (which is shown below to be the prior mean of γ_1). To explain the prior, let us first fix the scalar hyperparameter h_0 to zero, and discuss the first Gaussian kernel of (15), which corresponds to restrictions **C1** and **C2**. There, $\boldsymbol{\beta}_0$ is the prior expectation, and $\boldsymbol{Q}_0$ is the prior precision matrix. We specify this matrix to be diagonal:

$$\boldsymbol{Q}_0 = \text{diag}(0, 1/s_d^2, 1/s_a^2, \ldots, 1/s_a^2), \tag{16}$$

so that s_d is the prior standard deviation of the autoregressive coefficient and s_a is the prior standard deviation of the other coefficients. The strength with which restrictions **C1** and **C2** are imposed depends on the values of s_d and s_a, respectively: values close to zero correspond to a strong prior belief in favor of the restrictions. For the intercept term, the prior precision is set to zero, so that data information dominates the prior information on this term.

Although the prior expectation $\boldsymbol{\beta}_0$ embeds restriction **C3** that the sum of the last n elements of $\boldsymbol{\beta}$ is equal to 1, the prior variance of this sum is equal to $s_d^2 + (n-1)s_a^2$. Hence, to fix the latter variance to a small value, s_a itself must be fixed to an even smaller value, thus impacting how restriction **C2** is introduced. The second Gaussian kernel of (15) is designed to avoid the potential trade-off between the two restrictions, by adding a prior parameter

that controls the strength imposed on the unit sum, independently of the strength imposed on the individual coefficients. Notice that in the second exponential function of (15), we have written $\boldsymbol{\beta}_0'\boldsymbol{\iota}$ after the minus sign, instead of 1, to cover the case where one wants this target to be different from 1, that is, the case where one defines $\boldsymbol{\beta}_0$ differently from (10).

If $\boldsymbol{Q}_0$ in the first kernel is a null matrix, the second kernel specifies that the prior mean of the sum of the last n elements of $\boldsymbol{\beta}$ is equal to $\boldsymbol{\beta}_0'\boldsymbol{\iota}$ – i.e., equal to 1 if $\boldsymbol{\beta}_0$ is given by (10) – and that its prior precision is set to h_0. Hence a large value of h_0 corresponds to a strongly informative prior on the target value for the sum of the coefficients.

It is well-known that the product of two Gaussian kernels is a kernel of a Gaussian density. Hence, (15) is the kernel of the Gaussian density (see Section A of the appendix)

$$\boldsymbol{\beta} \sim \mathsf{N}_k(\boldsymbol{\beta}_0, \boldsymbol{V}_0), \tag{17}$$

where

$$\boldsymbol{V}_0 = (\boldsymbol{Q}_0 + h_0\boldsymbol{\iota}\boldsymbol{\iota}')^{-1}. \tag{18}$$

Notice that the expectation of $\boldsymbol{\beta}$ is $\boldsymbol{\beta}_0$, the same as in the first kernel in (15). If $h_0 > 0$, the prior covariance matrix is not diagonal: in fact, the covariances are negative, which is what is needed to reduce the prior standard deviation of $\boldsymbol{\beta}'\boldsymbol{\iota}$ compared to the value it takes when the prior covariance matrix is diagonal. Taking, for example, values that relate to the empirical illustration of Section 5, i.e., $d_0 = 0.5$, $s_d = s_a = 0.02$, $h_0 = 5000$, $n = 250$, then $\boldsymbol{\beta}_0 = (0, 0.5, 0.002008 \text{ (repeated 249 times)})$, $\boldsymbol{Q}_0^{-1} = \text{diag}(100, 0.02^2 \text{ (repeated 250 times)})$, the diagonal of $\boldsymbol{V}_0$ is $(100, 0.01996^2 \text{ (repeated 250 times)})$, the off-diagonal elements are equal to 0 in the first line (and column), and the other covariances are equal to $-1.59681/10^6$ (the corresponding correlation coefficient being equal to -0.004008). The prior standard deviation of $\boldsymbol{\beta}'\boldsymbol{\iota}$ is equal to 0.014128, i.e., much smaller than its corresponding value of 0.317 when $h_0 = 0$ and the prior is $\mathsf{N}_k(\boldsymbol{\beta}_0, \boldsymbol{Q}_0^{-1})$, where $\boldsymbol{Q}_0^{-1}$ is defined as $\text{diag}(0, s_d^2, s_a^2, \ldots, s_a^2)$.

To summarize, the prior density (17), when $\boldsymbol{\beta}_0$ is defined by (10) and $\boldsymbol{Q}_0$ by (16), is fully determined by the four scalar hyperparameters d_0, s_d, s_a, and h_0, whatever the dimension n of the VAR. These hyperparameters can be fixed to some values, as in the example above, or they can be chosen for each equation of the VAR by a cross-validation procedure similar to the procedure defined in the last paragraph of the previous subsection.

The computation of the posterior mean of $\boldsymbol{\beta}$ for the prior (14)-(17) is performed by a simple Gibbs sampling algorithm defined in Section B of the appendix. The prior is not conjugate since $\boldsymbol{V}_0$ is not proportional to σ^2. It becomes conjugate if (17) is replaced by

$$\boldsymbol{\beta}|\sigma^2 \sim \mathsf{N}_k(\boldsymbol{\beta}_0, \sigma^2\boldsymbol{V}_0). \tag{19}$$

The posterior mean corresponding to this conjugate prior is

$$\left(\boldsymbol{Z}'\boldsymbol{Z} + \boldsymbol{Q}_0 + h_0\boldsymbol{\iota}\boldsymbol{\iota}'\right)^{-1}\left(\boldsymbol{Z}'\boldsymbol{Y} + \boldsymbol{Q}_0\boldsymbol{\beta}_0 + h_0\boldsymbol{\iota}\boldsymbol{\iota}'\boldsymbol{\beta}_0\right), \tag{20}$$

where (18) has been used. If we set $\boldsymbol{Q}_0 = \boldsymbol{\Lambda}_k$ (by setting $\lambda_d^2 = 1/s_d^2$ and $\lambda_a^2 = 1/s_a^2$) and $h_0 = \lambda_s^2$, this posterior mean is exactly the ER estimator (13). With the non-conjugate prior,

one can only derive the conditional (to σ^2) posterior mean of $\boldsymbol{\beta}$, which can be expressed (see Section B of the appendix) as

$$\boldsymbol{\beta}_*(\sigma^2) = \left(\frac{\boldsymbol{Z}'\boldsymbol{Z}}{\sigma^2} + \boldsymbol{Q}_0 + h_0 \boldsymbol{\iota}\boldsymbol{\iota}' \right)^{-1} \left(\frac{\boldsymbol{Z}'\boldsymbol{Y}}{\sigma^2} + \boldsymbol{Q}_0\boldsymbol{\beta}_0 + h_0 \boldsymbol{\iota}\boldsymbol{\iota}'\boldsymbol{\beta}_0 \right). \tag{21}$$

This differs from (20) only by the presence of σ^2. The Gibbs sampler, also defined in Section B of the appendix, is a way to marginalize $\boldsymbol{\beta}_*(\sigma^2)$ with respect to σ^2. The resulting unconditional posterior mean of $\boldsymbol{\beta}$ then differs from the corresponding posterior mean/ER estimator when the prior is conjugate.

3.4 Forecasting

After obtaining a point estimate of $\boldsymbol{\beta}$ for an equation of the VAR system, such as the OLS estimator, the extended ridge estimator, or the posterior mean, a one-step ahead point forecast of y_{t+1} is simply obtained using a point estimate of (8) and the regressor $\boldsymbol{x}_t$. This is equivalent to using the point estimates of all equations to form the estimated $\boldsymbol{\tau}$ and $\boldsymbol{A}$ of the VAR system (7), and then computing one-step ahead point forecasts as $\hat{\boldsymbol{y}}_{t+1} = \hat{\boldsymbol{\tau}} + \hat{\boldsymbol{A}}\boldsymbol{y}_t$.

To compute h-step ahead forecasts, with $h > 1$, we can use either iterated multistep forecasting or direct multistep forecasting. An iterated h-step ahead forecast is based on the estimated VAR and computed recursively as $\hat{\boldsymbol{y}}_{t+h} = \hat{\boldsymbol{\tau}} + \hat{\boldsymbol{A}}\hat{\boldsymbol{y}}_{t+h-1}$. This approach amounts to computing $\hat{\boldsymbol{A}}^h$, i.e., to forecast all variables even if we are only interested in a subset of them (even just a single one). Hence, the forecast of a variable of interest may be contaminated by erroneous and imprecise forecasts of the other variables (see, e.g., Schorfheide, 2005).

If the objective is to forecast a subset of the series, or if one wishes to avoid the drawback inherent in the iterated multistep method highlighted above, the direct multistep forecasting method is preferable (see, e.g., Chevillon and Hendry, 2005, and Jordà, 2005). The method consists in directly projecting $\boldsymbol{y}_t$ on its lag $\boldsymbol{y}_{t-h}$, as in

$$\boldsymbol{y}_t = \boldsymbol{\tau}_h + \boldsymbol{A}_h\boldsymbol{y}_{t-h} + \boldsymbol{u}_{(\boldsymbol{h})\boldsymbol{t}}, \tag{22}$$

where for $h = 1$, $\boldsymbol{u}_{(\boldsymbol{1})\boldsymbol{t}} = \boldsymbol{\epsilon}_{\boldsymbol{t}}$ in (7). Ignoring that $\boldsymbol{A}_h = \boldsymbol{A}^h$, a typical equation of (22) can be cast in the form of (8) and (9), adapting the definitions of $\boldsymbol{Y}$, $\boldsymbol{x}_t$ and $\boldsymbol{Z}$. For $h > 1$, we denote the equation corresponding to (9) by

$$\boldsymbol{Y}_{(h)} = \boldsymbol{Z}_{(h)}\boldsymbol{\beta}_{(h)} + \boldsymbol{u}_{(h)}, \tag{23}$$

where $\boldsymbol{u}_{(h)}$ is a vector of errors. Although the elements of $\boldsymbol{u}_{(h)}$ are autocorrelated due to the recursive substitutions to get (22) from (7), we ignore this feature and we estimate the system (23) equation-by-equation, by OLS, ridge and Bayesian estimation, as is the case when $h = 1$. By proceeding in this spirit, no direct use is made in estimation of the relation $\boldsymbol{A}_h = \boldsymbol{A}^h$, because this would imply that the regression coefficients of the different equations of (22) are nonlinear functions of the same parameters (those of $\boldsymbol{A}$), so that equation-by-equation estimation would be pointless. In brief, the parameter $\boldsymbol{\beta}_{(h)}$ is not treated as a function of the underlying parameters of $\boldsymbol{A}$.

Yet, for ridge and Bayesian estimations, we allow the target towards which $\boldsymbol{\beta}_{(h)}$ is shrunk to be function of h, and we denote it by $\boldsymbol{\beta}_{(h),0}$. The target $\boldsymbol{\beta}_{(h),0}$ relates to the first row of $\boldsymbol{A}_0^h$, as in the case for $h = 1$, where $\boldsymbol{\beta}_0$ is directly the first row of $\boldsymbol{A}_0 = d_0\boldsymbol{I}_n + a_0\left(\boldsymbol{J}_n - \boldsymbol{I}_n\right)$, with $\boldsymbol{J}_n$ being a matrix of ones, and $a_0 = (1 - d_0)/(n - 1)$. In practice, we choose the last n elements of $\boldsymbol{\beta}_{(h),0}$ to be close to the first row of $\boldsymbol{A}_0^h$ when n is large relative to h: this is achieved by setting (see Section C of the appendix)

$$\boldsymbol{\beta}_{(h),0} = \left(0, d_0^h, \frac{1 - d_0^h}{n - 1}, \ldots, \frac{1 - d_0^h}{n - 1}\right)'. \tag{24}$$

The extended ridge estimator for the corresponding $\boldsymbol{\beta}_{(h)}$ is defined as in (13), replacing $\boldsymbol{\beta}_0$ with $\boldsymbol{\beta}_{(h),0}$, the penalty parameters and the value of d_0 being chosen by cross-validation for each horizon h. For Bayesian estimation, we use the same type of prior as when $h = 1$ (i.e., (14) and (17)), also replacing $\boldsymbol{\beta}_0$ with $\boldsymbol{\beta}_{(h),0}$. Forecasts for specific elements of $\boldsymbol{y}_t$ can readily be formed by estimating only specific rows of (22), so that forecasts are obtained from the corresponding individual equations, as in the case $h = 1$.

4 Monte Carlo Simulation

The results in CHL and Schennach do not imply that univariate long memory is necessarily due to the marginalization of a large system, only that it is possible. In this section, we explore via simulation whether the methods proposed above can shed light on this potential source of observed long memory. Owing to the use of different information sets for the univariate time series models and the AR-X models, we compare them via an out-of-sample forecasting competition. Our purpose is twofold: first, we show that, when long memory is not generated via a VAR(1) model satisfying the CHL or Schennach conditions but by univariate ARFIMA models, the methods proposed in Section 3 give worse out-of-sample forecasts than classical long memory models (such as the ARFIMA and HAR models), even if the series are correlated. Second, we find that, when the data are indeed generated by such a VAR(1) model, the use of our long-memory prone target or prior makes it possible to obtain forecasts as good as, or better than, the ARFIMA and HAR models – despite our use of only the first lag of a set of explanatory variables. More specifically, we compare, using simulated data displaying long memory, the forecasting performances of five estimation methods of the AR-X equations composing a VAR(1) system and three univariate time series models.

4.1 Simulation set-up

We consider two data generating processes (DGPs) and generate $T = 1,500$ observations for $n = 250$ series. DGP_1 is a set of n univariate ARFIMA$(0, d, 0)$ processes with $d = 0.45$ and correlated errors, i.e.,

$$(1 - L)^{0.45}\boldsymbol{y}_{n,t} = \boldsymbol{\Sigma}_n^{1/2}\boldsymbol{z}_{n,t}, \tag{25}$$

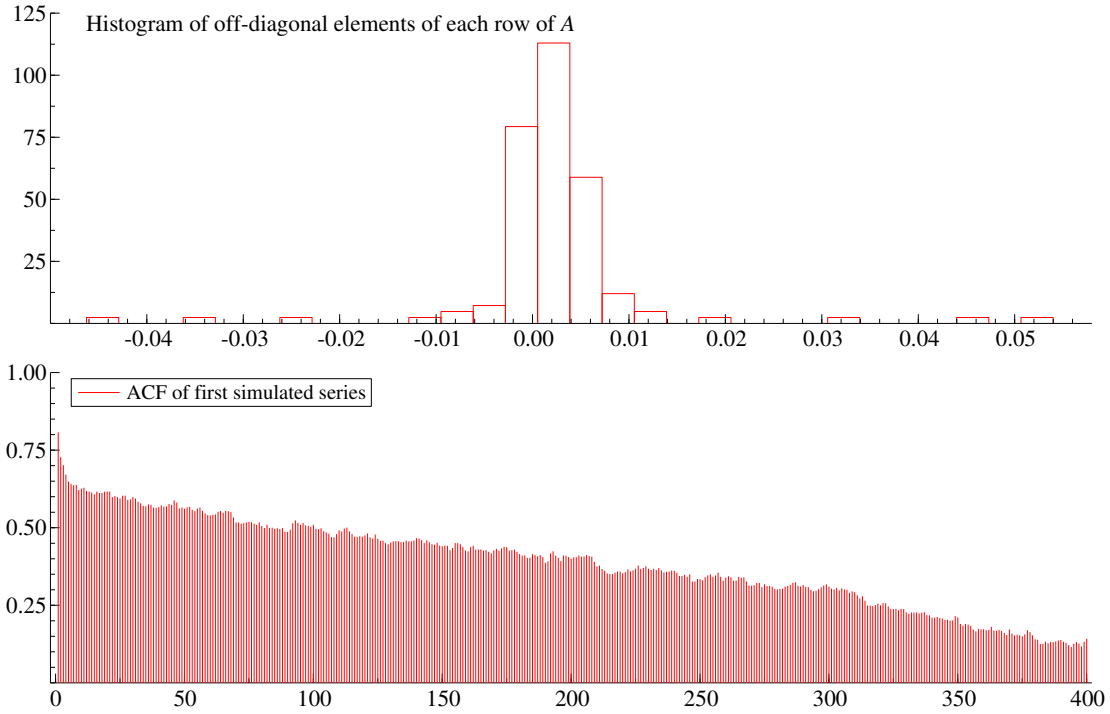

Figure 1: Top: Histogram of the values of the 249 off-diagonal elements of each row of the matrix $\boldsymbol{A}_n$ of the VAR(1) corresponding to DGP_2. Bottom: Autocorrelogram of the first simulated series.

where $\boldsymbol{z}_{n,t}$ is *i.i.d.* $\mathsf{N}(0, I_n)$ and $\boldsymbol{\Sigma}_n$ is an equicorrelation matrix with common correlation coefficient equal to 0.5, so that the n series form a system of correlated variables that display long memory.

DGP_2 is the VAR(1) system considered by CHL, i.e.,

$$(\boldsymbol{I}_n - \boldsymbol{A}_n L)\, \boldsymbol{y}_{n,t} = \boldsymbol{z}_{n,t}, \tag{26}$$

where $\boldsymbol{z}_{n,t}$ is defined as above and $\boldsymbol{A}_n$ is a Toeplitz matrix like (2). The entries of this matrix are defined as in CHL (their Example 1, page 56) where

$$t_k^{(n)} = \text{Re}\Big[\frac{1}{n}\sum_{j=0}^{n-1} g\big(\delta_n, e^{i\frac{2\pi j}{n}}\big) e^{-i\frac{2\pi jk}{n}}\Big], \tag{27}$$

where i is the imaginary unit, $\delta_n = 1/2 + o\left(n^{-2}\right)$ and $g(\delta, e^{i\omega})$ is defined, for $\delta \in (0, 1/2)$ and $\omega \in [0, 2\pi)$, as

$$g\big(\delta, e^{i\omega}\big) = 1_{\{0 \le \omega < \pi\delta\}} + 1_{\{\pi(\frac{3}{2} - \delta) < \omega \le \frac{3\pi}{2}\}}. \tag{28}$$

Since the entries of $\boldsymbol{A}_n$ defined in this way satisfy the properties (3a)-(3c), and $\boldsymbol{z}_{n,t}$ is i.i.d. Gaussian with diagonal covariance matrix, each element of $\boldsymbol{y}_{n,t}$ tends to a fractional white

noise process, as explained in Section 2. We set $\delta_n = 0.45$, so that in practice the diagonal elements of $\boldsymbol{A}_n$ are close to this value (0.452, precisely). In this respect the estimated degree of fractional integration of each generated series is essentially the same as in DGP_1. To shed light on DGP_2, the top graph of Figure 1 shows a histogram of the values of the off-diagonal elements of each row of the matrix $\boldsymbol{A}_n$ used for the data generation (the histogram is the same for each row due to the Toeplitz property of the matrix). It is clear that the values of these coefficients are close to, but different from, zero – illustrating (3b); the average of these values is equal to 0.0022008, and the sum of each row is equal to unity. The bottom graph illustrates the slow decay of the autocorrelation coefficients for one of the generated series.

It is clear that the source of long memory in DGP_1 differs from that of DGP_2. In DGP_1, the long memory originates in individual fractional integration, irrespective of the dependence across variables that is generated through correlations between the error terms (we do not report the unrealistic situation where all errors and variables are independent, but the results are qualitatively similar then). In DGP_2, the long range dependence is directly created by the dependence between the variables that is induced in the system through the features of matrix $\boldsymbol{A_n}$.

4.2 Competing models and set-up of the forecast comparison

We present below the set of eight competing models for a generic series y_t, i.e., an arbitrary element of $\boldsymbol{y}_{n,t}$.

1. AR: $y_t = \gamma_0 + \gamma_1 y_{t-1} + \epsilon_t$, estimated by OLS.

2. ARFIMA(1,d,0): $(1-L)^d(y_t - \gamma_0 - \gamma_1 y_{t-1}) = \epsilon_t$, estimated by Gaussian ML.

3. HAR (Corsi, 2009): $y_t = \gamma_0 + \gamma_1 y_{t-1} + \gamma_2 \frac{1}{5}\sum_{i=1}^{5} y_{t-i} + \gamma_3 \frac{1}{21}\sum_{i=1}^{21} y_{t-i} + \epsilon_t$, estimated by OLS.

4. AR-X: $y_t = \gamma_0 + \gamma_1 y_{t-1} + \sum_{i=2}^{n} \gamma_i x_{i,t-1} + \epsilon_t$, where $x_{i,t}$ for $i = 2, \ldots, n$ are the elements of $\boldsymbol{y}_{n,t}$ remaining after removing y_t. This is the model defined in (8). It is estimated by OLS.

5. RAR-X: this model is the AR-X estimated using the extended ridge estimator defined by (13), see Section 3.2. Recall that in this case we shrink γ_1 towards d_0 with penalty parameter λ_d^2, γ_i toward $(1-d_0)/(n-1)$ $(\forall i > 1)$ with penalty λ_a^2, and $\sum_{i=1}^{n} \gamma_i$ towards 1 with a penalty of λ_s^2. The penalty parameters (i.e., λ_d^2, λ_a^2 and λ_s^2) and d_0 are chosen by cross-validation as explained at the end of Section 3.2; details are provided in Section D of the appendix.

6. bRAR-X: this model is a basic version of the RAR-X, where all parameters (but the intercept) are shrunk towards 0 (i.e., a standard estimation of the model using the ridge technique). This is a special case of the method presented in Section 3.2, where $d_0 = a_0 = 0$, $\lambda_d^2 = \lambda_a^2 = \lambda^2$ and $\lambda_s^2 = 0$. λ^2 is chosen by cross-validation.

7. sRAR-X: this model is a strong version of the RAR-X model where the shrinkage targets are strongly imposed. More specifically, we impose $\gamma_1 = d_0$ and $\gamma_i = (1 - d_0)/(n-1)$ for $i = 2, \ldots, n$ and chose d_0 by cross-validation. Hence, this is the factor model version (6) of the AR-X equation.

8. BAR-X: this specification is also the AR-X but it is estimated by the Bayesian method presented in Section 3.3. The prior for the variance of ϵ_t is non-informative, see (14), and the prior for the regression coefficients $\boldsymbol{\beta} = (\gamma_0, \gamma_1, \gamma_2, \ldots, \gamma_n)'$ is the Gaussian density defined by (17) together with (10), (18) and (16). More specifically, the prior on γ_0 is quasi-noninformative (with a mean of 0 and a variance of 100), the prior mean of γ_1 is set equal to d_0, and the prior mean of γ_i, for all $i > 1$, is set to $(1-d_0)/(n-1)$. The prior precision of γ_1 is $1/s_d^2 + h_0$, the prior precision of γ_i $(i > 1)$ is $1/s_a^2 + h_0$. The co-precisions (the off-diagonal elements of the inverse of V_0) are all set to h_0. The larger h_0, the smaller the prior variance for the difference between the sum of the last n elements of $\boldsymbol{\beta}$ and the corresponding sum in the prior mean (equal to 1 for (10)). The prior parameters d_0, s_d, s_a and h_0 are chosen by cross-validation (see Section D of the appendix for details).

For models 5 to 8, cross-validation is carried out for each equation, so that the tuning parameters can differ from equation to equation.

If the data come from DGP_1, the AR-X-type of models are not nested in the DGP, whereas the ARFIMA(1,d,0) is a more general version of the DGP, the HAR is known to be a good approximation of this DGP while the AR model is a short memory process and therefore not able to capture long memory. Hence we expect the ARFIMA and HAR models to have better forecasting performances for DGP_1 than either the AR or the five AR-X models. By contrast, for DGP_2, we expect some of the AR-X-type of models estimated using the ER or Bayesian methods to perform at least as well as the ARFIMA and HAR, and the AR-X model estimated by OLS to underperform because of in-sample over-fitting.

Each competing model is used to produce h-step ahead forecasts (for $h = 1, 5$ and 10). For $h > 1$, we use iterated multistep forecasts (i.e., recursive substitution) for AR, ARFIMA and HAR, and direct multistep forecasts for AR-X, the three RAR-X and BAR-X. As discussed in Subsection 3.4, this avoids contaminating forecasts across variables when additional (non autoregressive) regressors are present.

The eight models are estimated, for each of the 250 available series, on rolling windows of $T_0 = 1,000$ observations. We start with the initial estimation window ($t = 1, ..., T_0$), and keeping the same parameters estimates, we compute 25 consecutive h-step ahead forecasts ($h = 1, 5, 10$). Next, we shift the estimation window forward by 25 observations (to the subsample $t = 26, ..., T_0 + 25$) and re-estimate all parameters, which we use to produce 25 consecutive h-step ahead forecasts. This procedure is continued until the last estimation window, $t = 476, ..., T_0 + 475$. This results in a number of 500-h individual h-step ahead forecasts.

To speed up the estimations, the tuning parameters of the three RAR-X and BAR-X models are only selected once by cross-validation on the first window of $T_0 = 1,000$ observations and then kept constant.

The out-of-sample forecasts are compared to the observed values using the mean squared forecast error (MSE) loss functions defined for each model m and horizon h as

$$MSE_h^{(m)} = \frac{1}{T_h} \sum_{t=1}^{T_h} (\hat{y}_{t,h}^{(m)} - y_t)^2, \tag{29}$$

where T_h is the number of forecasts at horizon h, and $\hat{y}_{t,h}^{(m)}$ is the forecast of y_t at horizon h by model m.

The comparison tool is the model confidence set (MCS) procedure of Hansen, Lunde, and Nason (2011), see Section D of the appendix for details about the implementation. Note that for both DGPs, the variance of the error term is equal to one for each series so that we expect the MSE of well-performing models to be close to 1 for $h = 1$.

4.3 Simulation results

The results of the simulations are reported in Table 1 for three forecast horizons. We only present results for one simulated dataset since this involves comparing $250 \times (500 - h + 1)$ forecasts for each model and horizon h, i.e., 371,750 forecasts in total for each model – yet the results are essentially identical when we use other seeds for the random number generators.

For DGP_1, ARFIMA, closely followed by HAR, has the lowest average MSEs and the highest proportions of series included in the MCS at the 95% level. The other models are much less effective. AR is unable to capture long memory; bRAR-X is shrinking all slope parameters to zero (including the AR(1) coefficient) and does not improve much compared to the AR. AR-X has too many parameters relative to the number of observations and therefore suffers from a problem of over-fitting. The sRAR-X model specifies long memory too tightly in the direction of CHL and Schennach (which are distinct from the DGP), while RAR-X and BAR-X do the same but less tightly so: their average MSE is approximately ten percent higher than those of ARFIMA and HAR.

For DGP_2, sRAR-X, RAR-X and BAR-X have the lowest average MSEs and the highest proportions of series included in the MCS at the 95% level, at the three forecast horizons. These models are designed to exploit the multivariate asymptotic mechanism of the DGP which generates long memory; this is more the case for sRAR-X than for RAR-X and BAR-X, hence the excellent performance of sRAR-X is not surprising, while the very good one of RAR-X and BAR-X is reassuring. ARFIMA and HAR do not use the multivariate asymptotic mechanism of the DGP which generates long memory, but they are able to capture the long memory in the data, although less systematically than the models that are congruent with the DGP. AR and AR-X perform as badly as for DGP_1, for the same reasons. That bRAR-X is as good as RAR-X and BAR-X at horizons 5 and 10 is the single surprising result of this simulation.

The conclusions from this simulation exercise are that our long memory prone target proposals and associated estimation methods appear well designed and that they possess good model discovery properties. Indeed, their success at forecasting in the context of a long-memory DGP based on CHL (DGP_2) contrasted with their lower performance when

Table 1: Results of the Monte Carlo simulations

h	AR	ARFIMA	HAR	AR-X	bRAR-X	sRAR-X	RAR-X	BAR-X
				$\underline{\text{DGP}_1\text{: ARFIMA, eq (25)}}$				
				Frequencies at which each model belongs to the 95%-MCS				
1	0.092	1.000	0.932	0.000	0.028	0.000	0.148	0.144
5	0.244	0.996	0.920	0.000	0.316	0.008	0.448	0.416
10	0.316	0.984	0.876	0.012	0.396	0.008	0.512	0.460
				Average MSE				
1	1.131	1.026	1.033	1.459	1.281	1.282	1.119	1.119
5	1.828	1.468	1.484	2.402	1.764	2.109	1.622	1.628
10	2.019	1.616	1.641	2.366	1.878	2.407	1.770	1.777
				$\underline{\text{DGP}_2\text{: VAR, eq (26)-(28) with } \delta = 0.45}$				
				Frequencies at which each model belongs to the 95%-MCS				
1	0.148	0.680	0.636	0.000	0.172	0.988	0.964	0.908
5	0.400	0.604	0.464	0.000	0.900	1.000	0.828	0.896
10	0.412	0.772	0.384	0.000	0.896	1.000	0.852	0.900
1	0.004	0.576	0.592	0.000	0.112	0.988	0.924	0.896
5	0.000	0.456	0.428	0.000	0.868	1.000	0.816	0.896
10	0.000	0.452	0.372	0.000	0.844	0.988	0.828	0.892
				Average MSE				
1	1.114	1.054	1.062	1.363	1.151	1.024	1.026	1.026
5	1.490	1.409	1.455	1.930	1.327	1.311	1.323	1.321
10	1.516	1.406	1.491	1.971	1.353	1.338	1.349	1.347
1	1.172	1.060	1.061	1.367	1.158	1.024	1.028	1.028
5	2.007	1.445	1.456	1.922	1.329	1.308	1.320	1.320
10	2.212	1.467	1.489	1.946	1.340	1.321	1.332	1.331

The models are defined in Section 4.2. The top parts of each panel report the frequencies (over the 250 series) at which each model belongs to the MCS at the 95% confidence level (i.e., the proportions of p-values above 5% in the MCS procedure) for a given forecasting horizon h, an MSE loss function and over the full forecasting period of $500 - h + 1$ observations. The bottom parts of each panel record the corresponding average MSE (over the 250 series).

the source of long memory is idiosyncractic rather than systemic (as is the case with DGP_1) shows that the performance of the proposed methods constitutes a good gauge for eliciting the multivariate source of long memory of CHL and Schennach. We turn next to assessing this property empirically.

5 Empirical illustration

In this section, we present an application to data where long memory has been well documented in the literature, for which a large number of variables assumed to belong to the same system are available, and where long time series can be obtained. For these three reasons we chose to focus on the logarithm of a measure of daily volatility for a set of 250 U.S. company stocks.

5.1 Tracing the source of the long memory in log realized volatility

The initial dataset (purchased from tickdatamarket) consists of transaction prices at the 1-second sampling frequency for 1,412 stocks from the NYSE, AMEX and NASDAQ markets, for the period ranging from January 1st, 1991 to October, 14, 2019 covering 7,510 trading days. We ordered the stocks by decreasing average daily transaction volume, and kept the 250 largest capitalization stocks for the period from 2005-01-03 to 2017-07-24 (3,276 trading days). These start and end dates were chosen to maximize the number of available series out of the larger dataset of 7,510 trading days.

We aggregated the data at the 5-minute frequency and computed the MedRV estimator of Andersen, Dobrev, and Schaumburg (2012), a non-parametric robust-to-jumps estimator of the integrated variance. If $r_{t,i}$ is the ith 5-minute return of a given stock on a day t containing $M = 78$ (since trading is from 9:30 to 16) such returns, $\log(MedRV_t)$ (denoted by y_t hereafter) is computed as the logarithm of

$$MedRV_t = \frac{\pi}{6 - 4\sqrt{3} + \pi} \frac{M}{M-2} \sum_{i=3}^{M} \text{median}(|r_{t,i}|, |r_{t,i-1}|, |r_{t,i-2}|)^2.$$

The presence of long memory in the volatility of the log-returns of financial assets is a well recognized stylized fact (see Baillie, Bollerslev and Mikkelsen, 1996, Breidt, Crato and de Lima, 1998, Comte and Renault, 1998, Andersen, Bollerslev, Diebold and Ebens, 2001, among others), and numerous models that have been proposed generate long memory in volatility (e.g., Giraitis, Robinson and Surgailis, 2000, Hurvich, Moulines and Soulier, 2005, or Lieberman and Phillips, 2008, who focus specifically on realized volatility). There is however a debate on whether the short-run autoregressive parameter in an ARFIMA(1,d,0) for the logarithm of realized volatility may be near-unity so the corresponding d would be close to -1/2 to generate the same long range dependence, see Shi and Yu (2022). Multivariate techniques, VAR and Vector Heterogenous Autoregressive (VHAR) models, have been used for forecasting realized volatility measures by Anderson and Vahid (2007) and Cubadda, Hecq, and Riccardo (2019), respectively.

For our data, the average value (over the 250 series) of the estimated d parameters of the ARFIMA$(1, d, 0)$ model obtained on the full sample is 0.48, with a standard deviation of 0.02. We estimated the same eight competing models as in Section 4.2, again on rolling windows of $T_0 = 1,000$ observations. The first estimation sample spans the period from 2005-01-03 to 2008-10-31, and h-step ahead forecasts of y_t are computed for several horizons ($h = 1, 2, \ldots, 10, 20, 30, 40, 50$) leading to a total number of $2,277$ minus h forecasts. As in Section 4, we rely on the MCS procedure of Hansen et al. (2011) and a MSE loss function to rank the models in terms of their forecasting performance.

Table 2 reports the results of the MCS procedure applied to the full out-of-sample period. More specifically the figures in this table report for each forecasting horizon h the frequencies (over the 250 series) at which each model belongs to the MCS at the 95% confidence level (or equivalently the percentage of p-values higher than 5%). The second part of the table reports the corresponding relative average (over the 250 series) MSE of each model with respect to the RAR-X model, i.e. the ratio of the value for a model to the value for RAR-X. Results for the AR-X-type of models with iterative forecasts (as explained in Section 3.4) are not reported since their predictive performance is much lower than when forecasts are obtained by the direct multistep forecasting method.

We notice the following results in Table 2:

- AR, AR-X, bRAR-X and sRAR-X are strongly outperformed by the other models over the full forecasting period. Their average losses (see the bottom panel of Table 2) are larger (often much more so) than those of other models. The AR, AR-X and sRAR-X models are almost never included in the MCS, while bRAR-X performs better than these three models for $h \leq 6$.

- ARFIMA and HAR perform similarly, especially in terms of average losses. Their frequencies of inclusion in the MCS are also close, except for horizons above 10, where HAR underperforms compared to ARFIMA. Overall, these frequencies fluctuate between 14.8 and 44 percent.

- RAR-X and BAR-X perform better than all the other models in terms of average losses and also generally for the MCS inclusion rates – for which, RAR-X actually beats BAR-X always except at $h = 1$. Furthermore, like HAR, the performance of BAR-X deteriorates for $h > 10$. Overall, the best model is RAR-X. This model belongs to the MCS for between 48.8 and 76 percent of the cases.

In brief, the use of the theoretical constraints in the AR-X model through the proposed Bayesian and ridge estimation methods strongly improves the model forecasting performance with respect to OLS. The bad performance of the latter is due to a lack of precision because 251 coefficients are estimated using 1,000 observations, whereas the shrinkage methods impose a relevant theoretical structure on the estimated coefficients. The performance of the shrinkage methods is on average markedly superior to that of the ARFIMA and HAR models; this difference can be attributed to the use of a larger, but relevant, information set.

Table 2: Results of the MCS test and average MSE values for h-step-ahead forecasts

h	AR	ARFIMA	HAR	AR-X	bRAR-X	sRAR-X	RAR-X	BAR-X
Frequencies at which each model belongs to the 95%-MCS								
1	0.000	0.424	0.388	0.000	0.108	0.000	0.488	0.544
2	0.000	0.384	0.360	0.000	0.076	0.000	0.700	0.552
3	0.000	0.384	0.364	0.004	0.152	0.000	0.700	0.456
4	0.000	0.396	0.364	0.004	0.184	0.000	0.604	0.536
5	0.000	0.424	0.364	0.000	0.260	0.000	0.556	0.504
6	0.000	0.416	0.396	0.000	0.216	0.000	0.608	0.548
7	0.000	0.420	0.400	0.000	0.012	0.000	0.648	0.480
8	0.000	0.440	0.436	0.000	0.008	0.000	0.676	0.528
9	0.000	0.408	0.380	0.000	0.012	0.004	0.636	0.556
10	0.000	0.352	0.340	0.000	0.004	0.000	0.664	0.540
20	0.000	0.336	0.284	0.000	0.016	0.000	0.760	0.276
30	0.000	0.316	0.236	0.000	0.004	0.004	0.748	0.364
40	0.000	0.316	0.156	0.000	0.012	0.004	0.668	0.304
50	0.000	0.332	0.148	0.000	0.000	0.012	0.636	0.284
Average MSE of model in column header, relative to RAR-X								
1	1.245	1.047	1.051	1.285	1.126	1.221	1.000	0.998
2	1.391	1.089	1.096	1.289	1.092	1.418	1.000	1.007
3	1.496	1.095	1.103	1.278	1.074	1.539	1.000	1.007
4	1.554	1.077	1.085	1.260	1.060	1.575	1.000	0.997
5	1.594	1.062	1.071	1.253	1.041	1.574	1.000	0.989
6	1.645	1.064	1.071	1.247	1.084	1.575	1.000	0.991
7	1.695	1.073	1.079	1.252	1.249	1.579	1.000	1.000
8	1.724	1.071	1.077	1.251	1.246	1.565	1.000	0.998
9	1.751	1.073	1.080	1.249	1.246	1.558	1.000	0.995
10	1.773	1.080	1.087	1.248	1.246	1.551	1.000	0.997
20	1.762	1.088	1.098	1.268	1.266	1.437	1.000	1.046
30	1.756	1.144	1.161	1.290	1.287	1.428	1.000	1.017
40	1.687	1.154	1.183	1.331	1.328	1.379	1.000	1.011
50	1.665	1.188	1.224	1.358	1.350	1.369	1.000	1.009

The models are defined in Section 4. The top part reports the frequencies (over the 250 series) at which each model belongs to the MCS at the 95% confidence level (i.e., the proportions of p-values above 5% in the MCS procedure) for a given forecasting horizon h, an MSE loss function and over the full forecasting period of $2277 - h$ observations The bottom part reports the ratios of the average MSE (over the 250 series) of each model to that of RAR-X.

5.2 Subsample stability

To check if the empirical results for the full sample are not specific to the chosen forecasting period, we study their subsample stability.

First, we report in Figure 2 the evolution over the sample of the average (over the 250 series) cumulative sum of squared (forecast) errors (CSSE) of each model divided by the CSSE of the RAR-X model (with a burning period of 21 days). The figure shows that the rankings of the models are very stable through the forecasting period. The values at the last date correspond to those reported in Table 2; for example, at horizon 50, the MSE of ARFIMA is 18.8% larger than the MSE of RAR-X on the full forecasting period.

Second, we also apply the MCS procedure on rolling windows of 250 forecasts., and report the results graphically, see Figure 3, More specifically, the graphs on the left-hand side panels of the figure show, for three representative forecast horizons ($h = 1, 10$ and 50) the averages (over the 250 stocks) of the MSE loss functions computed over rolling windows of 250 forecasts. The corresponding panels on the right-hand side of the figure show the time evolutions of the frequencies at which each model belongs to the MCS at the confidence level of 95%. A frequency of 50 (percent) for model m at date t means that the model m is in the MCS for fifty percent of the 250 series, the MCS in question being obtained using the loss function computed from the 250 forecasts ending at date t. Note that to reduce the computing time, the MCS procedure is not applied to every consecutive window of 250 forecasts, but to every twenty-fifth window, so that each line is drawn by joining 82 values. The full-sample rankings of model forecasting performances – as analyzed in the previous subsection – remain valid over most subsamples, but it can be seen that ARFIMA or HAR occasionally perform better than BAR-X or RAR-X. The differences between RAR-X/BAR-X and ARFIMA/HAR are more pronounced at horizon $h = 50$ than at shorter horizons.

Overall, the results above show that full-sample dominance of RAR-X and BAR-X methods over ARFIMA and HAR constitutes a feature that is reflected in most of the subsamples.

5.3 Impact of the cross-sectional dimension

The results presented so far used the maximum possible number of variables in the AR-X equations, i.e., the $n = 250$ available series. To assess the theoretical models of CHL and Schennach where long memory is an asymptotic feature, we can vary the number of regressors in the AR-X equations. Indeed, if the data exhibit long memory due to marginalization of a large system, the RAR-X and BAR-X model performances should be improving when the number of regressors is increased. Hence, we carry out estimations and forecasts for the 250 series, keeping always the idiosyncratic autoregressive term, but changing the number of "X" regressors, i.e., the lagged realized volatilities of the other assets. We denote by K the total number of regressors in the equation and compare the results for the corresponding RAR-X model. We only focus on the latter as it is the best forecasting model according to the results presented in Table 2.

The results are presented in Table 3 for six values of K and four forecast horizons. The six models estimate different numbers of parameters using, each, a different target, $\boldsymbol{A}_K^0$ in

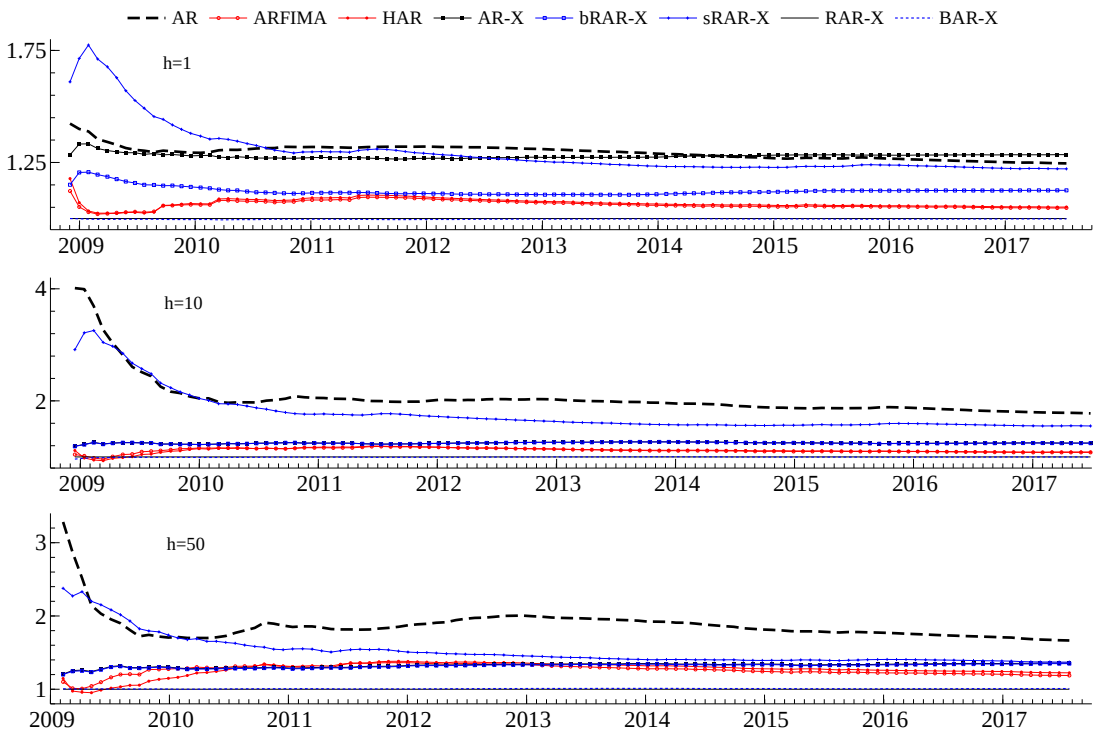

Figure 2: Evolution of the ratio of the cumulative sum of squared forecast errors of each model with respect to the RAR-X model, over the sample period, for three forecast horizons ($h = 1, 10, 50$). The models are the same as reported in Table 2.

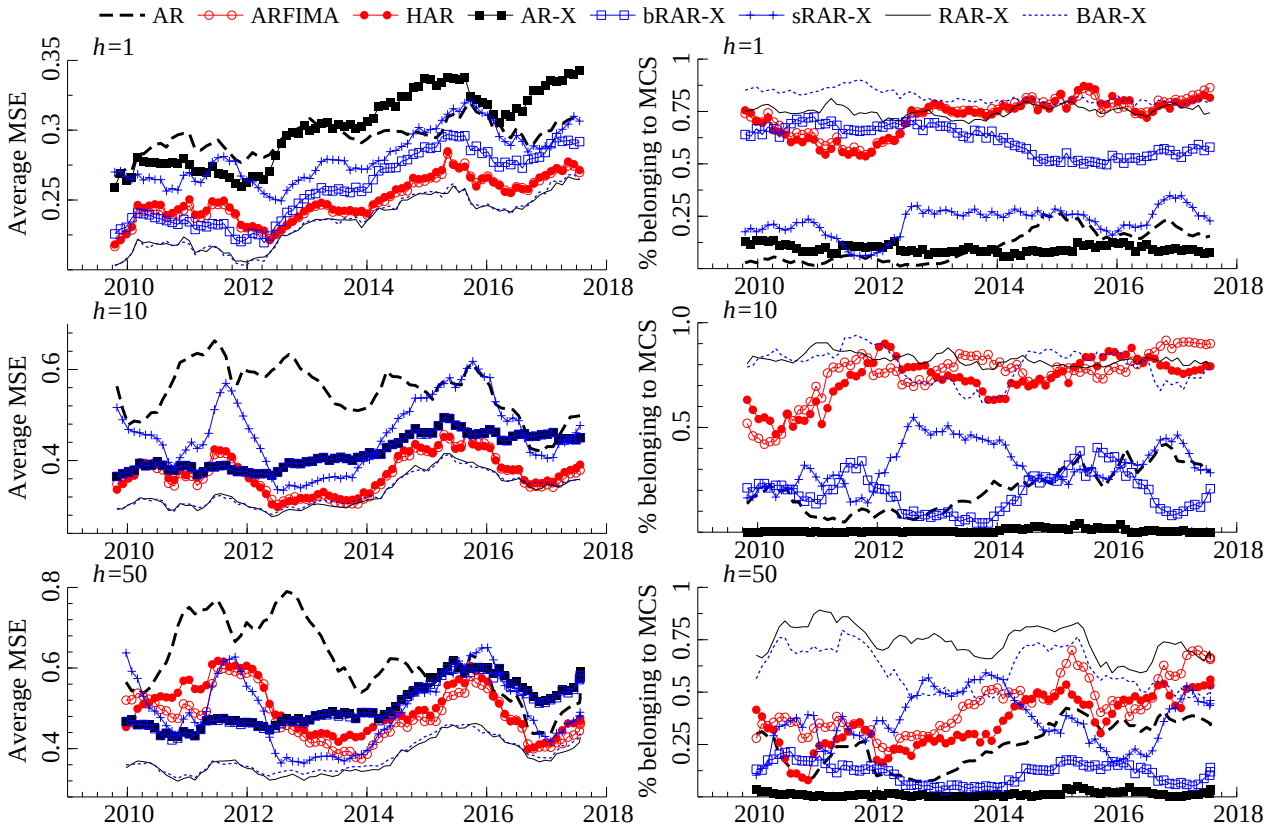

Figure 3: Left panels: Average (over the 250 series) MSE loss computed on rolling windows of 250 forecasts. Right panels: Frequencies (over the 250 series) at which each model belongs to the MCS (at the 95% confidence level) for the MSE loss function and rolling windows of 250 forecasts. The top graph is for forecast horizon $h = 1$, the middle one for $h = 10$, and the bottom one for $h = 50$.

equation (5). The MCS frequencies for $K = 250$ in Table 3 differ from those in Table 2 because the sets of models of the MCS procedure differ: for Table 3, the model set includes only the RAR-X models for the six selected values of K.

Table 3: MCS frequencies and average MSE values for the RAR-X model, for an increasing number of regressors (K)

h	$K = 10$	$K = 50$	$K = 100$	$K = 150$	$K = 200$	$K = 250$
Frequencies at which model K belongs to 95%-MCS						
1	0.008	0.080	0.184	0.380	0.528	0.828
5	0.016	0.060	0.212	0.372	0.536	0.832
10	0.028	0.060	0.192	0.412	0.564	0.864
50	0.252	0.220	0.192	0.344	0.520	0.760
Average MSE of model K, relative to model with $K = 250$						
1	1.167	1.065	1.035	1.016	1.006	1.000
5	1.264	1.110	1.053	1.021	1.008	1.000
10	1.267	1.129	1.062	1.026	1.012	1.000
50	1.305	1.120	1.068	1.042	1.010	1.000

The RAR-X model is defined in Section 4, item 5. K is the number of regressors, not counting the constant term.

Increasing the number of regressors decreases the average MSE, and this effect is stronger for small values of K than for large values. By contrast, while increasing K also increases the frequency at which the RAR-X model for a given K is included in the MCS, the latter effect is stronger at larger values of K. We can conclude that it is worth using a large number of lagged realized volatilities of other assets to improve the forecasts, in line with the theoretical predictions of CHL and Schennach.

5.4 In-sample residual serial correlation

Although the BAR-X and especially RAR-X models, as implemented so far, forecast better than their univariate time series competitors (ARFIMA, HAR), one may ask (as raised by a referee) whether the in-sample residuals of these models (i.e., the residuals over the estimation period) are exempt from autocorrelation. If it is not the case, this could indicate that the models do not fully capture in-sample the long memory observed in the data, or at least that they do not perform in this respect as well as their time series competitors.

For each model and each of the 250 series of in-sample residuals, we therefore computed Box-Pierce statistics of order 20, denoted by BP(20). By choosing an order of 20 instead of a very small or very large value, we should be able to conclude if long memory is captured (if the statistic is insignificant) or not (if it is significant). We prefer to refer to the BP(20) statistic rather than an estimate of the degree of residual long memory as it is a more

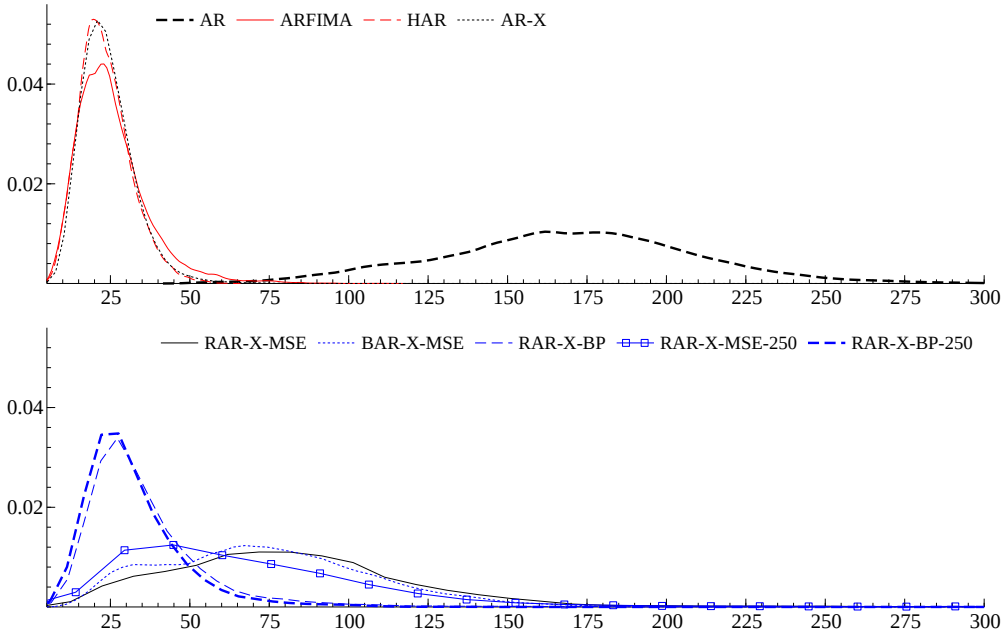

Figure 4: Empirical distributions over 250 variables of BP(20) statistics computed for in-sample residuals. Each line corresponds to one of the models (estimated at horizon $h = 1$) presented in Table 4 (see the corresponding table note for definitions).

direct measure that residuals follow the canonical assumption of no serial correlation. For reference below, note that an asymptotic, 5% sized, chi-squared test based on the BP(20) statistic rejects the null of no residual autocorrelation (until order 20) for realizations greater than 31.4.

We consider the residuals of the eight models presented in Table 2 and complement them with three models that we explain progressively below. To synthesize this information, Figure 4 shows the empirical distributions of the 250 BP(20)-statistics for nine models. Two models (bBAR-X and sRAR-X) are not reported in the graph as the corresponding distributions show that they do not correct the in-sample residuals for serial correlation better than BAR-X and RAR-X. The nine distributions are shown on two comparable graphs to enhance their visibility.

It is obvious in Figure 4 that the AR model fails, for all series, to clean the residuals of autocorrelation. BAR-X-MSE and RAR-X-MSE (denoted previously, and respectively, BAR-X and RAR-X in Table 2) also present a large proportion (approximately 90%) of BP(20) statistics above the 31.4 critical value. By contrast, for ARFIMA, HAR, and AR-X, the proportion of statistics below 31.4 are 77%, 85%, and 84% respectively. As regards AR-X (estimated by OLS), its good performance – as good as HAR – is explained by overfitting caused by the estimation of a large number of parameters (n) relative to the number of

observations (T_0), since overfitting produces zero residuals in the limiting case where $n = T_0$.

Hence, it is clear that the best out-of-sample forecasting models do not clean the in-sample autocorrelation of the data as well as the time-series models. To investigate this issue further, we consider two additional empirical strategies.

> **Regularly updated cross-validation:** instead of choosing the tuning parameters by cross-validation over a first training sample of 1000 observations, and keeping them fixed for the whole forecasting exercise, we renew regularly the tuning parameters after updating the training sample using a rolling window scheme, by removing the 250 observations at the beginning of the previous training sample and adding 250 new observations; we call this a '250-rolling-scheme'.
>
> **Box-Pierce cross-validation:** instead of choosing the tuning parameters on the basis of the MSE values of the forecast period of the training sample, we choose them based on the BP(20)-statistics of the residuals of the training sample.

We therefore add three models to the eight models presented in Table 2; the three new models are also included in Figure 4. We report the forecasting results for the extended set of eleven models in Table 4. The three new models are:

- RAR-X-MSE-250: the RAR-X model with "Regularly updated cross-validation" (i.e., the '250-rolling-scheme' defined above) based on the MSE criterion;
- RAR-X-BP: the RAR-X model with "Box-Pierce cross-validation" computed for the initial training sample only;
- RAR-X-BP-250: the RAR-X model with "Box-Pierce" and "Regularly updated" cross-validation (i.e., based jointly on the BP(20)-statistic and the '250-rolling-scheme').

To distinguish BP from MSE cross-validation, we add an "-MSE" suffix when this criterion is used. We do not consider the BAR-X-BP, BAR-X-MSE-250 and BAR-X-BP-250 models because the required computing time was beyond our computing resources and also because RAR-X globally outperforms BAR-X so far.

Regarding the average MSE values, we note that, since the values in the first eight models are the same in Tables 2 and 4, the interesting point is to compare them with those of the new models (i.e., those in the last three columns of Table 4). We see that, while RAR-X-BP-250 has the smallest average MSE for almost all horizons, the five RAR-X and BAR-X models (in the last five columns of the table), with different cross-validation schemes (including the previously identified best models, RAR-X-MSE and BAR-X-MSE) perform very similarly, with very small differences in average MSEs.

When comparing the models on the basis of their frequencies of inclusion in the MCS, we find more important differences, namely that:

Table 4: Results of the MCS test and average MSE values, with three additional models

h	AR	ARFIMA	HAR	AR-X	bRAR-X -MSE	sRAR-X -MSE	RAR-X -MSE	BAR-X -MSE	RAR-X -MSE-250	RAR-X -BP	RAR-X -BP-250
Frequencies at which each model belongs to the 95%-MCS											
1	0.000	0.404	0.368	0.000	0.016	0.000	0.400	0.396	0.628	0.524	0.568
2	0.000	0.384	0.364	0.000	0.056	0.000	0.572	0.536	0.604	0.572	0.676
3	0.000	0.376	0.368	0.000	0.100	0.000	0.632	0.424	0.464	0.532	0.556
4	0.000	0.364	0.356	0.000	0.104	0.000	0.556	0.436	0.448	0.672	0.636
5	0.000	0.412	0.380	0.000	0.172	0.000	0.504	0.420	0.416	0.580	0.592
6	0.000	0.396	0.368	0.000	0.140	0.000	0.536	0.420	0.460	0.560	0.660
7	0.000	0.408	0.384	0.000	0.004	0.000	0.556	0.412	0.444	0.608	0.704
8	0.000	0.420	0.400	0.000	0.000	0.000	0.556	0.428	0.432	0.648	0.720
9	0.000	0.388	0.360	0.000	0.004	0.000	0.544	0.448	0.472	0.656	0.688
10	0.000	0.348	0.360	0.000	0.000	0.000	0.576	0.452	0.484	0.688	0.708
20	0.000	0.288	0.268	0.000	0.000	0.000	0.604	0.164	0.556	0.640	0.696
30	0.000	0.308	0.244	0.000	0.000	0.004	0.616	0.328	0.492	0.572	0.684
40	0.000	0.308	0.172	0.000	0.000	0.004	0.516	0.224	0.500	0.516	0.524
50	0.000	0.332	0.152	0.000	0.000	0.008	0.468	0.228	0.436	0.424	0.480
Average MSE of model in column header, relative to RAR-X-BP-250											
1	1.252	1.053	1.057	1.292	1.132	1.228	1.006	1.004	0.994	1.006	1.000
2	1.401	1.097	1.104	1.299	1.100	1.429	1.008	1.015	1.013	1.008	1.000
3	1.514	1.108	1.116	1.293	1.086	1.557	1.012	1.019	1.019	1.005	1.000
4	1.594	1.105	1.113	1.292	1.088	1.616	1.026	1.023	1.026	1.001	1.000
5	1.644	1.095	1.104	1.293	1.074	1.624	1.031	1.020	1.028	1.002	1.000
6	1.698	1.098	1.106	1.288	1.120	1.626	1.033	1.024	1.030	1.002	1.000
7	1.741	1.102	1.108	1.286	1.282	1.621	1.027	1.026	1.030	1.002	1.000
8	1.774	1.102	1.109	1.288	1.282	1.610	1.029	1.027	1.034	1.002	1.000
9	1.802	1.105	1.112	1.286	1.282	1.604	1.029	1.024	1.031	1.000	1.000
10	1.823	1.110	1.117	1.284	1.281	1.595	1.028	1.026	1.032	1.000	1.000
20	1.810	1.117	1.128	1.303	1.301	1.477	1.027	1.074	1.021	1.000	1.000
30	1.768	1.152	1.169	1.299	1.295	1.438	1.007	1.024	1.009	1.001	1.000
40	1.685	1.153	1.182	1.329	1.326	1.377	0.999	1.010	0.993	1.000	1.000
50	1.640	1.170	1.205	1.337	1.330	1.348	0.985	0.993	0.984	1.001	1.000

The models with MSE in their name (columns 5-8) are the same as the models in columns 5-8 of Table 2. RAR-X-MSE-250, RAR-X-BP and RAR-X-BP-250 are defined in Subsection 5.4. All other models are defined in Section 4.2.

1. The BP criterion for cross-validation improves upon MSE: RAR-X-BP has larger frequencies than RAR-X-MSE for 9 horizons (among which 6 differences are larger than 5 percentage points), and RAR-X-BP-250 has larger frequencies than RAR-X-MSE-250 for 13 horizons (among which 11 differences are larger than 5 points);

2. Updating the training sample regularly (the '250-rolling-scheme') versus using only one training sample is beneficial when using the BP criterion for the cross-validations: RAR-X-BP-250 has higher frequencies than RAR-X-BP for 13 horizons (among which 7 differences are larger than 5 points). By contrast, when the criterion is MSE, RAR-X-MSE-250 has lower frequencies than RAR-X-MSE for 12 horizons (among which 9 differences are larger than 5 points).

From these comparisons, we conclude that, although the average MSEs do not vary much with respect to the cross-validation criterion, the best forecasting model is the RAR-X-BP-250. It therefore appears preferable to use the BP criterion for cross-validation, especially when the training sample is updated regularly, as we implemented (with a '250-rolling-scheme').

Looking back at Figure 4, we see that the distributions of the BP(20)-statistics for RAR-X-BP-250 and RAR-X-BP are much closer to those of ARFIMA and HAR than the distributions for RAR-X-MSE and RAR-X-MSE-250.

The interesting lesson of this section is that the BP criterion improves both the out-of-sample forecasts and the in-sample residuals.

6 Conclusions

This paper considers a novel approach for the empirical modeling of variables exhibiting long memory, using one lag of a large cross-section of related variables instead of the usual technique that models variables using a long history of their own lags. This approach is based on two theoretical contributions that prove that long memory in a variable can be caused by its dependences within a large system or network. We provide two estimation methods that harness the informativeness of the theoretical models and use them to drive the estimation, either via an extended ridge regression that shrinks the estimates toward a structure derived from the theory, or by using the latter to design an informative prior in a Bayesian setup.

In an application to realized volatilities of stocks, we show that the proposed modeling and estimation strategy improves upon standard univariate models (between 5% and 20% reduction in mean square forecast error compared to ARFIMA and HAR models, depending on the forecast horizon) in terms of predicting series characterized by the presence of long memory. Such results suggest that it may be fruitful to model variables that exhibit long range dependence by using one lag of a set of related variables, provided that the cross-sectional dimension is large. Among the other possible empirical applications with these potential properties, we may think for instance of sectorial consumer price indices, bond yields, river streamflows, or input-output production networks.

The success of the proposed approach opens the door to more exploration of the impact that dependences within a large network or system may have on each variable's idiosyncractic long range persistence. It could be extended to include richer short-term dynamics (e.g., through VAR(p) modeling, with $p > 1$) or higher dimensional, e.g., spatial, networks.

Appendix

A: Proof of (13) and of (17)

Proof of (13): notice that $(\boldsymbol{\beta}'\boldsymbol{\iota}-\boldsymbol{\beta}_0'\boldsymbol{\iota})^2 = (\boldsymbol{\beta}'\boldsymbol{\iota}-\boldsymbol{\beta}_0'\boldsymbol{\iota})(\boldsymbol{\beta}'\boldsymbol{\iota}-\boldsymbol{\beta}_0'\boldsymbol{\iota})' = \boldsymbol{\beta}'\boldsymbol{\iota}\boldsymbol{\iota}'\boldsymbol{\beta}-2\boldsymbol{\beta}'\boldsymbol{\iota}\boldsymbol{\iota}'\boldsymbol{\beta}_0+\boldsymbol{\beta}_0'\boldsymbol{\iota}\boldsymbol{\iota}'\boldsymbol{\beta}_0$. By developing the quadratic forms, the ER objective function (12) is equal to $\boldsymbol{\beta}'\boldsymbol{Z}'\boldsymbol{Z}\boldsymbol{\beta} - 2\boldsymbol{\beta}'\boldsymbol{Z}'\boldsymbol{Y}+\boldsymbol{\beta}'\boldsymbol{\Lambda}_k\boldsymbol{\beta}-2\boldsymbol{\beta}'\boldsymbol{\Lambda}_k\boldsymbol{\beta}_0+\lambda_s^2\boldsymbol{\beta}'\boldsymbol{\iota}\boldsymbol{\iota}'\boldsymbol{\beta}-2\lambda_s^2\boldsymbol{\beta}'\boldsymbol{\iota}\boldsymbol{\iota}'\boldsymbol{\beta}_0+\boldsymbol{Y}'\boldsymbol{Y}+\lambda\boldsymbol{\beta}_0'\boldsymbol{\Lambda}_k\boldsymbol{\beta}_0+\lambda_s^2\boldsymbol{\beta}_0'\boldsymbol{\iota}\boldsymbol{\iota}'\boldsymbol{\beta}_0$. Solving the first-order condition yields the solution (13).

Proof of (17): to show that the kernel (15) corresponds to (17), we can write that (15) is equal to

$$\exp\{-\frac{1}{2}[(\boldsymbol{\beta}-\boldsymbol{\beta}_0)'\boldsymbol{Q}_0(\boldsymbol{\beta}-\boldsymbol{\beta}_0)+h_0(\boldsymbol{\beta}'\boldsymbol{\iota}-\boldsymbol{\beta}_0'\boldsymbol{\iota})(\boldsymbol{\beta}'\boldsymbol{\iota}-\boldsymbol{\beta}_0'\boldsymbol{\iota})']\} = K_0\exp[-\frac{1}{2}f(\boldsymbol{\beta})],$$

where K_0 does not depend on $\boldsymbol{\beta}$ and

$$f(\boldsymbol{\beta}) = \boldsymbol{\beta}'(\boldsymbol{Q}_0+h_0\boldsymbol{\iota}\boldsymbol{\iota}')\boldsymbol{\beta}-2\boldsymbol{\beta}'(\boldsymbol{Q}_0\boldsymbol{\beta}_0+h_0\boldsymbol{\iota}\boldsymbol{\beta}_0'\boldsymbol{\iota}) = (\boldsymbol{\beta}-\bar{\boldsymbol{\beta}}_0)'\boldsymbol{V}_0{}^{-1}(\boldsymbol{\beta}-\bar{\boldsymbol{\beta}}_0)+C_0,$$

where $\boldsymbol{V}_0{}^{-1} = \boldsymbol{Q}_0+h_0\boldsymbol{\iota}\boldsymbol{\iota}'$, $\bar{\boldsymbol{\beta}}_0 = \boldsymbol{V}_0(\boldsymbol{Q}_0\boldsymbol{\beta}_0+h_0\boldsymbol{\iota}\boldsymbol{\beta}_0'\boldsymbol{\iota})$, and $C_0 = \bar{\boldsymbol{\beta}}_0'\boldsymbol{V}_0^{-1}\bar{\boldsymbol{\beta}}_0$ does not depend on $\boldsymbol{\beta}$. Hence, the prior density depends on $\boldsymbol{\beta}$ only through $\exp[-\frac{1}{2}(\boldsymbol{\beta}-\bar{\boldsymbol{\beta}}_0)'\boldsymbol{V}_0{}^{-1}(\boldsymbol{\beta}-\bar{\boldsymbol{\beta}}_0)]$, which is the kernel of the Gaussian density $N_k(\bar{\boldsymbol{\beta}}_0,\boldsymbol{V}_0)$. To show that this Gaussian density is the same as (17), we show that $\bar{\boldsymbol{\beta}}_0 = \boldsymbol{\beta}_0$:

$$\begin{aligned}
\bar{\boldsymbol{\beta}}_0 &= (\boldsymbol{Q}_0+h_0\boldsymbol{\iota}\boldsymbol{\iota}'^{-1}(\boldsymbol{Q}_0\boldsymbol{\beta}_0+h_0\boldsymbol{\iota}\boldsymbol{\beta}_0'\boldsymbol{\iota}) = (\boldsymbol{Q}_0^{-1}-\frac{h_0\boldsymbol{Q}_0^{-1}\boldsymbol{\iota}\boldsymbol{\iota}'\boldsymbol{Q}_0^{-1}}{1+h_0\boldsymbol{\iota}'\boldsymbol{Q}_0^{-1}\boldsymbol{\iota}})(\boldsymbol{Q}_0\boldsymbol{\beta}_0+h_0\boldsymbol{\iota}\boldsymbol{\beta}_0'\boldsymbol{\iota}) \\
&= \boldsymbol{\beta}_0+h_0\boldsymbol{Q}_0^{-1}\boldsymbol{\iota}\boldsymbol{\beta}_0'\boldsymbol{\iota}-\frac{1}{1+h_0\boldsymbol{\iota}'\boldsymbol{Q}_0^{-1}\boldsymbol{\iota}}(h_0\boldsymbol{Q}_0^{-1}\boldsymbol{\iota}\underbrace{\boldsymbol{\iota}'\boldsymbol{Q}_0^{-1}\boldsymbol{Q}_0\boldsymbol{\beta}_0}_{=\boldsymbol{\beta}_0'\boldsymbol{\iota}}+h_0\boldsymbol{Q}_0^{-1}\boldsymbol{\iota}\boldsymbol{\iota}'\boldsymbol{Q}_0^{-1}h_0\boldsymbol{\iota}\boldsymbol{\beta}_0'\boldsymbol{\iota}) \\
&= \boldsymbol{\beta}_0+h_0\boldsymbol{Q}_0^{-1}\boldsymbol{\iota}\boldsymbol{\beta}_0'\boldsymbol{\iota}\Big(1-\frac{1}{1+h_0\boldsymbol{\iota}'\boldsymbol{Q}_0^{-1}\boldsymbol{\iota}}-\frac{h_0\boldsymbol{\iota}'\boldsymbol{Q}_0^{-1}\boldsymbol{\iota}}{1+h_0\boldsymbol{\iota}'\boldsymbol{Q}_0^{-1}\boldsymbol{\iota}}\Big) = \boldsymbol{\beta}_0.
\end{aligned}$$

In the first line, the explicit form of the inverse of $\boldsymbol{Q}_0+h_0\boldsymbol{\iota}\boldsymbol{\iota}'$ is obtained by applying the Sherman-Morrison formula.

B: Bayesian estimation of the AR(1)-X model

The results exposed in this appendix are included for ease of reference. They are well known, see e.g., Bauwens, Lubrano, and Richard (1999) for details.

For the regression Equation (9), with the assumption of normality of the error term, the prior (14) and (17), the posterior density of $\boldsymbol{\beta}$ and σ^2 is proportional to

$$(\sigma^2)^{-(T+2)/2}\exp\{-\frac{\hat{s}}{2\sigma^2}\}\exp\{-\frac{1}{2}(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}})'\frac{\boldsymbol{Z}'\boldsymbol{Z}}{\sigma^2}(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}})\}\exp\{-\frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\beta}_0)'\boldsymbol{V}_0^{-1}(\boldsymbol{\beta}-\boldsymbol{\beta}_0)\}, \quad (30)$$

where $\hat{\boldsymbol{\beta}}$ is the OLS estimator $(\boldsymbol{Z}'\boldsymbol{Z})^{-1}\boldsymbol{Z}'\boldsymbol{Y}$, and $\hat{s}$ is the sum of squared OLS residuals.

Because the prior density is not conjugate, the posterior marginal density of $\boldsymbol{\beta}$ is not available analytically. However, the posterior density of $(\boldsymbol{\beta}, \sigma^2)$ can be simulated by applying a Gibbs sampler iterating between $\boldsymbol{\beta}$ and $\boldsymbol{\sigma}^2$. Indeed, the posterior density of $\boldsymbol{\beta}$ conditional on σ^2 is Gaussian:

$$\boldsymbol{\beta}|\sigma^2, \boldsymbol{Y}, \boldsymbol{Z} \sim \mathsf{N}_k(\boldsymbol{\beta}_*, \boldsymbol{V}_*), \quad (31)$$

where

$$\boldsymbol{V}_* = \left(\frac{\boldsymbol{Z}'\boldsymbol{Z}}{\sigma^2} + \boldsymbol{V}_0^{-1}\right)^{-1}, \quad (32)$$

$$\boldsymbol{\beta}_* = \boldsymbol{V}_*\left(\frac{\boldsymbol{Z}'\boldsymbol{Y}}{\sigma^2} + \boldsymbol{V}_0^{-1}\boldsymbol{\beta}_0\right) := \boldsymbol{\beta}_*(\sigma^2). \quad (33)$$

and the complementary conditional density of σ^2 is inverted-gamma:

$$\sigma^2|\boldsymbol{\beta} \sim \mathsf{IG}(T, (\boldsymbol{Y}-\boldsymbol{Z}\boldsymbol{\beta})'(\boldsymbol{Y}-\boldsymbol{Z}\boldsymbol{\beta})). \quad (34)$$

The Gibbs sampling algorithm to generate S draws $(\boldsymbol{\beta}^{(s)}, (\sigma^2)^{(s)})$, for $s = 1, 2, \ldots, S$, from the posterior of the parameters (after S_0 warming-up draws) is organized as follows:

1. *Choose an initial value* $(\sigma^2)^{(0)}$ *(e.g.* $\hat{s}/(T-k-2)$*).*
2. *Set* $s = 1$.
3. *Draw successively* $\boldsymbol{\beta}^{(s)}$ *from* the normal density (31) where β_* and Q_* are computed with $\sigma^2 = (\sigma^2)^{(s-1)}$, and $(\sigma^2)^{(s)}$ *from* $\mathsf{IG}(T, (\boldsymbol{Y}-\boldsymbol{Z}\boldsymbol{\beta}^{(s)})'(\boldsymbol{Y}-\boldsymbol{Z}\boldsymbol{\beta}^{(s)}))$.
4. *Set* $s = s+1$ *and go to step 3 unless* $s > S_0 + S$.
5. *Discard the first* S_0 *values of* $\boldsymbol{\beta}^{(s)}$ *and* $(\sigma^2)^{(s)}$.

The posterior expectation of $\boldsymbol{\beta}$ is approximated by the mean of the S draws $\boldsymbol{\beta}^{(s)}$, or by the mean of the S conditional expectations $\boldsymbol{\beta}_*[(\sigma^2)^{(s)}]$.

C: Explanation of (24)

Using $a_0 = (1-d_0)/(n-1)$, $\boldsymbol{A}_0 = d_0\boldsymbol{I}_n + a_0(\boldsymbol{J}_n - \boldsymbol{I}_n) = \frac{nd_0-1}{n-1}\boldsymbol{I}_n + \frac{1-d_0}{n-1}\boldsymbol{J}_n$. Using $\boldsymbol{J}_n^h = n^{h-1}\boldsymbol{J}_n$ for $h \geq 1$ and denoting $\boldsymbol{I}_n$ as $\boldsymbol{J}_n^0$,

$$\begin{aligned}\boldsymbol{A}_0^h &= \sum_{j=0}^{h} \frac{h!}{j!\,(h-j)!}\left[\left(\frac{nd_0-1}{n-1}\right)^{h-j}\left(\frac{1-d_0}{n-1}\right)^{j}\right]\boldsymbol{J}_n^j \\ &= \left(d_0 + \frac{d_0-1}{n-1}\right)^h \boldsymbol{I}_n + \frac{1}{n}\left[1 - \left(d_0 + \frac{d_0-1}{n-1}\right)^h\right]\boldsymbol{J}_n\end{aligned}$$

and hence $\boldsymbol{A}_0^h = \left(d_0^h + o\left(n^{-1}\right)\right)\boldsymbol{I}_n + \left(\frac{1-d_0^h}{n} + o\left(n^{-1}\right)\right)\boldsymbol{J}_n$, for $n >> h$, so that the first row is then close to $\left(d_0^h, \frac{1-d_0^h}{n}, ..., \frac{1-d_0^h}{n}\right)'$. The target $\boldsymbol{\beta}_{(h),0}$ in (24) is obtained by putting 0 as first element and dividing the last $n-1$ elements by $n-1$ (instead of n) to ensure that the sum of the target is exactly equal to 1. The reason why we prefer this target to using exactly the first row of $\boldsymbol{A}_0^h$ is that it ensures the autoregressive coefficient does not depend on the dimension of the system.

D: Technical details

Model confidence set

The procedure of Hansen et al. (2011) is applied using the MSE loss function defined in (29) to perform the hypothesis tests of equal predictive accuracy needed to obtain each model confidence set. Let $\mathcal{M}_0$ be the set of competing models. The relative performance is measured by $d_{i,j,t} = L_{i,t} - L_{j,t}$ for all $i, j \in \mathcal{M}_0$. The MCS test is an iterative procedure. For iteration s, it applies a model equivalence test for the null hypothesis of

$$H_{0,\mathcal{M}_s} : E\left(d_{ij,t}\right) = 0 \text{ for all } i, j \in \mathcal{M}_s \subset \mathcal{M}_0,$$

against the alternative

$$H_{A,\mathcal{M}_s} : E\left(d_{ij,t}\right) \neq 0 \text{ for some } i, j \in \mathcal{M}_s.$$

If $H_{0,\mathcal{M}_s}$ is 'accepted' the confidence set $\hat{\mathcal{M}}_{1-\alpha} = \mathcal{M}_s$, otherwise use an elimination rule to remove objects from $\mathcal{M}^s$ and repeat the test. Let $P_{H_{0,\mathcal{M}_s}}$ be the p-value associated with the null hypothesis $H_{0,\mathcal{M}_s}$ and $e_{\mathcal{M}_s}$ be the model eliminated from set $\mathcal{M}_s$ when $H_{0,\mathcal{M}_s}$ is rejected. The MCS p-value for model $e_{\mathcal{M}_s}$ is defined by

$$\hat{p}_{e_{\mathcal{M}_s}} = \max_{k \leq s} P_{H_{0,\mathcal{M}_k}},$$

where $\mathcal{M}_1 \supset \mathcal{M}_2 \ldots \supset \mathcal{M}_s$.

The MCS test is performed at the 5% significance level, so that the resulting MCS is at the confidence level of 95%. The test statistic is the range statistic that requires a bootstrap procedure. 1,000 bootstrap samples are used, with a block length of 5 observations to account for potential serial correlation and conditional heteroscedasticity in the losses.

Cross validation

Table 5 reports the grids of the cross validations performed to choose the values of the tuning parameters that determine the shrinkage of the RAR(1)-X and BAR(1)-X models for the simulations and the application.

Table 5: Grids for the cross validations

RAR(1)-X	d_0	0.2 to 0.55 by steps of 0.025
	λ_d^{-1}	0.01 to 0.05 by steps of 0.01
	λ_a^{-1}	0.01 to 0.05 by steps of 0.01
	λ_s^2	0 to 5,000 by steps of 1,000
BAR(1)-X	d_0	0.2 to 0.55 by by steps of 0.05
	s_d	0.01 to 0.05 by steps of 0.01
	s_a	0.01 to 0.05 by steps of 0.01
	h_0	0 to 5,000 by steps of 1,000

Figure 5 provides the histograms of the values obtained by the cross validations, for RAR-X and BAR-X and $h = 1$. The ordinates show the number of series, for example d_0 is equal to 0.55 for a little less than 150 series (out of 250) for RAR-X and a little more than 150 for BAR-X.

The parameters $1/\lambda_d$ of RAR-X and s_d in BAR-X are selected at the lowest values of the grid (0.01 or 0.02) for about two-thirds of the series. The parameters $1/\lambda_a$ of RAR-X and s_a in BAR-X are selected differently between RAR-X and BAR-X.

The additional shrinkage of the sum of the coefficients toward 1 by the parameter λ_s^2 (RAR-X) or the equivalent parameter h_0 (BAR-X) is effective for abound 120 series (about 48 percent). The impact of the shrinkage of the sum towards unity is, however, effective through the other constraints. The OLS estimated sum ranges from -0.06 to 1.50 (over the 150 series), the mean being 0.92 and the standard deviation 0.23; RAR-X estimation results in the range (0.66, 1.44), with mean 0.97 and standard deviation 0.11; the BAR-X range is (0.65, 1.48) with the same mean and standard deviation as RAR-X.

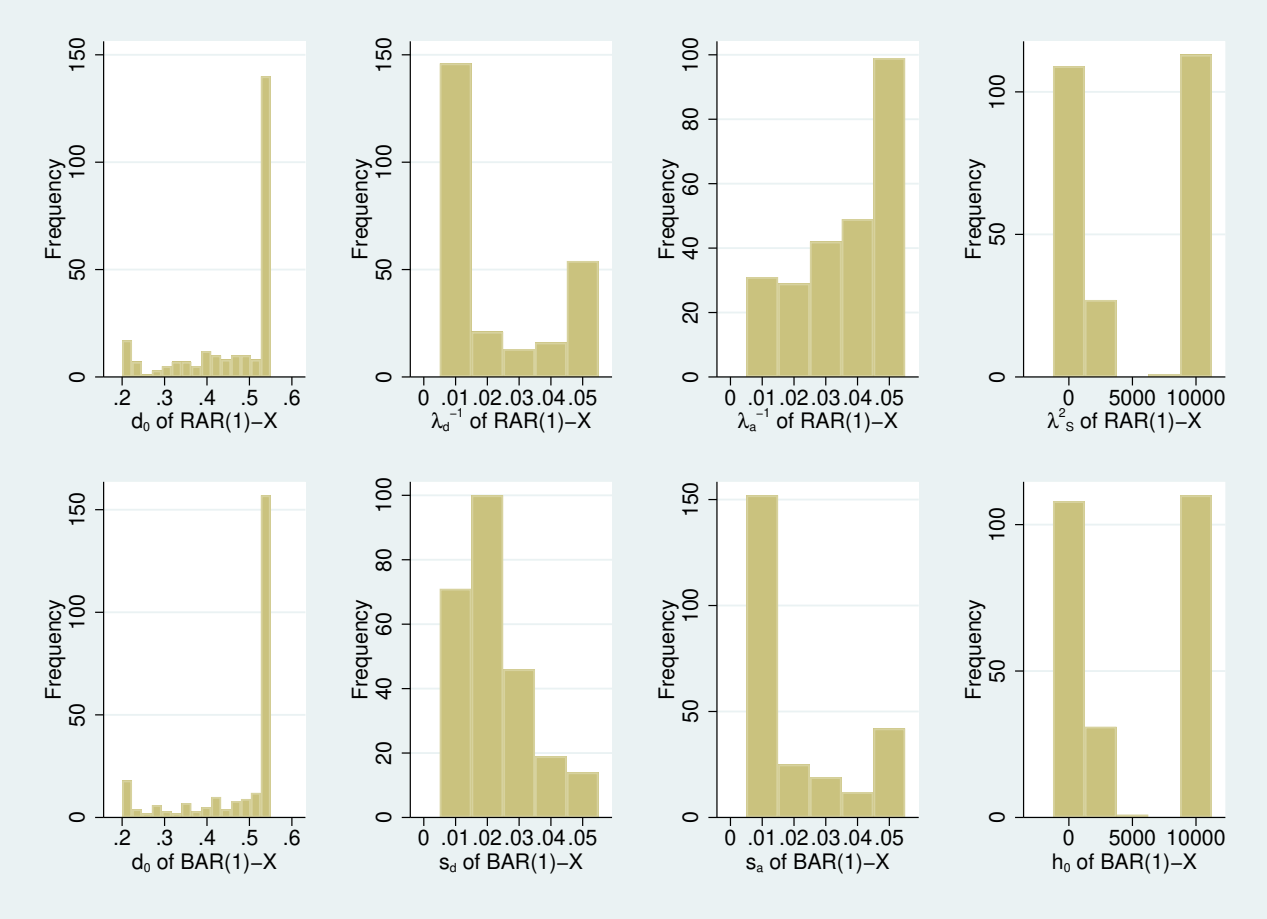

Figure 5: Histogram of the four tuning parameters estimated by cross validation on the first sample of 1,000 observations for the application.

References

Andersen, T., D. Dobrev, and E. Schaumburg (2012). Jump robust volatility estimation using nearest neighbor truncation. *Journal of Econometrics 169*(1), 75–93.

Andersen, T. G., T. Bollerslev, F. X. Diebold, and H. Ebens (2001). The distribution of realized stock return volatility. *Journal of financial economics 61*(1), 43–76.

Anderson, H. M. and F. Vahid (2007). Forecasting the volatility of Australian stock returns: Do common factors help? *Journal of Business & Economic Statistics 25*(1), 76–90.

Baillie, R., T. Bollerslev, and H. Mikkelsen (1996). Fractionally integrated generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics 74*, 3–30.

Bauwens, L., M. Lubrano, and J.-F. Richard (1999). *Bayesian Inference in Dynamic Econometric Models*. Oxford University Press.

Beran, J. (1992). Statistical methods for data with long-range dependence. *Statistical Science 7*(4), 404–416.

Breidt, F. J., N. Crato, and P. de Lima (1998). The detection and estimation of long memory in stochastic volatility. *Journal of Econometrics 83*(1-2), 325–348.

Chevillon, G., A. Hecq, and S. Laurent (2018). Generating univariate fractional integration within a large VAR(1). *Journal of Econometrics 204*(1), 54–65.

Chevillon, G. and D. F. Hendry (2005). Non-parametric direct multi-step estimation for forecasting economic processes. *International Journal of Forecasting 21*(2), 201–218.

Chudik, A. and M. H. Pesaran (2011). Infinite-dimensional VARs and factor models. *Journal of Econometrics 163*(1), 4–22.

Comte, F. and E. Renault (1998). Long memory in continuous-time stochastic volatility models. *Mathematical Finance 8*, 291–323.

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

Cubadda, G., A. Hecq, and A. Riccardo (2019). Forecasting realized volatility measures with multivariate and univariate models: The case of the us banking sector. In J. Chevallier, S. Goutte, G. D., S. Saglio, and B. Sanhaji (Eds.), *Financial Mathematics, Volatility and Covariance Modelling*, Volume 2, Chapter 11, pp. 791–897. Routledge.

Diebold, F. X. and K. Yılmaz (2009). Measuring financial asset return and volatility spillovers, with application to global equity markets. *The Economic Journal 119*, 158–171.

Diebold, F. X. and K. Yılmaz (2014). On the network topology of variance decompositions: Measuring the connectedness of financial firms. *Journal of Econometrics 182*, 119–134.

Doan, T., R. Litterman, and C. Sims (1984). Forecasting and conditional projection under realistic prior distributions. *Econometric Reviews 3*, 1–100.

Giannone, D., M. Lenza, and G. Primiceri (2019). Priors for the long run. *Journal of the American Statistical Association 114*, 565–580.

Giraitis, L., P. M. Robinson, and D. Surgailis (2000). A model for long memory conditional heteroscedasticity. *Annals of Applied Probability 10*(3), 1002–1024.

Granger, C. W. J. and R. Joyeux (1980). An introduction to long-memory time series models and fractional differencing. *Journal of time series analysis 1*(1), 15–29.

Hansen, P., A. Lunde, and J. Nason (2011). The model confidence set. *Econometrica 79*, 453–497.

Harvey, A. C. (1990). *Forecasting, structural time series models and the Kalman filter*. Cambridge University Press.

Hurvich, C. M., E. Moulines, and P. Soulier (2005). Estimating long memory in volatility. *Econometrica 73*(4), 1283–1328.

Jordà, Ò. (2005). Estimation and inference of impulse responses by local projections. *American economic review 95*(1), 161–182.

Karlsson, S. (2013). Forecasting with Bayesian vector autoregressions. In G. Elliot and A. Timmermann (Eds.), *Handbook of Economic Forecasting*, Volume 2, Chapter 15, pp. 791–897. Elsevier.

Lieberman, O. and P. C. Phillips (2008). Refined inference on long memory in realized volatility. *Econometric reviews 27*(1-3), 254–267.

Müller, U. K. and M. W. Watson (2008). Testing models of low-frequency variability. *Econometrica 76*(5), 979–1016.

Pesaran, M. H. (2006). Estimation and inference in large heterogeneous panels with a multifactor error structure. *Econometrica 74*(4), 967–1012.

Schennach, S. M. (2018). Long memory via networking. *Econometrica 86*(6), 2221–2248.

Schorfheide, F. (2005). VAR forecasting under misspecification. *Journal of Econometrics 128*(1), 99–136.

Shi, S. and J. Yu (2022). Volatility puzzle, forthcoming in *Management Science*.