Bayesian Compressed Vector Autoregressions

Gary Koop[a], **Dimitris Korobilis**[b], and Davide Pettenuzzo[c]

[a]University of Strathclyde
[b]University of Glasgow
[b]Brandeis University

2015 NBER-NSF Time Series Conference
Vienna, 25-26 September, 2015

Large VARs

- Vector autoregressions (VARs) is an important tool in applied macroeconomics since Sims (1980)
- Recently, many researchers define large VARs involving dozens of dependent variables
- E.g. Banbura, Giannone and Reichlin (2010); Carriero, Kapetanios and Marcellino (2009); Koop and Korobilis (2013); Korobilis (2013); Giannone, Lenza, Momferatou and Onorante (2014)
- Typically these large models have many more parameters than observations
- And typical solution is to use shrinkage methods: PCA; Minnesota (Bayes) priors; LASSO etc

What we do in this paper

- We build on ideas from the machine learning literature and develop a VAR model in the form of a "compressed regression"
- Compressed sensing/compressive sampling are methods for shrinking large dimensional data
- In the machine learning literature this would involve millions of data (not so common in economics)
- We focus on forecasting macro VARs $\rightarrow$ e.g. Banbura et al (2010) estimate 132-variable VARs with 13 lags (200,000 parameters)
- Unlike Principal Component Analysis, the compressed regression methods we use are "supervised" (will explain this in due course)
- First such modelling attempt in economics, preliminary results show good forecasting performance in BVARs

Random Projection

- The main idea we build upon is that of “Random Projection” (RP)
- High-dimensional data is projected onto a low-dimensional subspace using a random matrix, whose columns have unit length
- “loadings” (projection) matrix is not estimated from data, rather generated randomly, e.g. N(0,1)
- There exist theoretical results supporting that RP preserves for example volumes and affine distances, or the structure of data (e.g., clustering)
- The central idea of RP is based on the Johnson-Lindenstrauss lemma
- Lemma implies that if we perform an orthogonal projection of *n* points in a vector space onto a selected lower-dimensional subspace, then distances between points are preserved

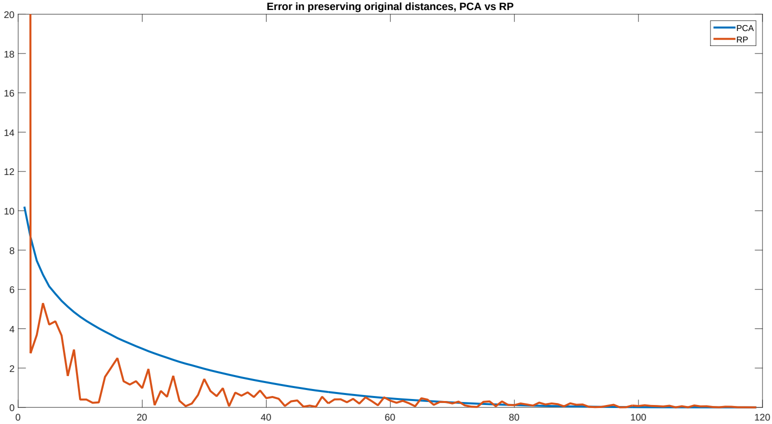
Comparison with other methods

- RP is a projection method similar to Principal Component Analysis (PCA)
- Unlike PCA, RP does not depend on a particular training data set
- Unlike Discrete Cosine Transform (DCT) or Discrete Fourier Transform (DFT) its basis vectors do not exhibit particular frequency or phase properties
- RP doesn’t compute a low-dimensional subspace by optimizing certain criteria, thus is data independent
- Sometimes mentioned as a “Data Oblivious” method
- Inexpensive in terms of time/space, and can be generated without even seeing the data
- Therefore, computationally simple method which can be used in “big-data”

Intuition behind RPs 1

- With orthogonal RPs distances between two points of the original data are preserved
- Assume a large matrix of data x, and a projection matrix R (but do NOT think of a parametric model yet, such as DFM, VAR etc).
- The Euclidean distance of two points in the original space is $\|x_i - x_j\|$, while in the projected space this is $\sqrt{\frac{d}{k}}\|Rx_i - Rx_j\|$, where d is the original & k reduced dimensionality of data.
- To get a first idea, the next figure plots the error from the distance $\|Rx_i - Rx_j\|$ compared to the original distance $\|x_i - x_j\|$
- The matrix x has 119 columns (macro variables) for 600+ monthly obs, and we take the average distances of all possible combinations of pairs x_i, x_j from this matrix
- We compare two methods, i) R is generated randomly; ii) R is estimated optimally using PCA

PCA vs RP


Error in preserving original distances, PCA vs RP
PCA
RP
0
2
4
6
8
10
12
14
16
18
20
0
20
40
60
80
100
120

VAR Methodology

VAR(p) for $M \times 1$ vector of dependent variables is :

$$y_t = \alpha_0 + \sum_{j=1}^{p} A_j y_{t-j} + \varepsilon_t, \varepsilon_t \sim N(0, \Sigma) \tag{1}$$

This model can be written compactly

$$y_t = A^{'} X_t + \varepsilon_t \tag{2}$$

- X_t is $K \times 1$, and K can be large
- E.g. A VAR with $M = 100$ variables and $p = 12$ lags has $K = 120,000+$
- While there are so many methods for high-dimensional data, motivation of our approach is based on computational considerations

Methods to estimate large VARs

- large VARs have large X and possibly large y
- Two major solutions for dealing with large VARs:
    1. Specify large VAR for y and X and shrink coefficients (e.g. LASSO, MCMC variable selection)
    2. Shrink either y and/or X and estimate smaller system (e.g. PCA, reduced rank VAR)
- In solution 1, $(X'X)^{-1}$ can be slow to compute, especially repeatedly (e.g. MCMC, Monte Carlo, Bootstrap)
- Therefore, methods that shrink parameters have, in general, limitations (e.g. $M < 40 - 50$)
- Subjective Minnesota prior using analytical results is one exception (Banbura, Giannone, Reichlin 2010, JAE)
- Alternative is solution 2: project data to lower dimensions
- We may lose ability to unfold structural economic relationships

Our approach: incorporate RPs in the VAR

Following Guhaniyogi and Dunson (2015, JASA), we define the following Compressed VAR

$$y_t = B^{'} \left(\Phi X_t \right) + \varepsilon_t \tag{3}$$

- Projection matrix Φ is $m \times K$, $m \ll K$
- We generate Φ randomly (explain schemes in next slide)
- Conditional on knowing Φ, VAR above is trivial to estimate $\rightarrow B$ is of lower dimensions
- Will explain now how we generate Φ, decide on m, and why the method works

Generation of Projection matrix

The matrix Φ can be generated quickly, e.g. using the scheme

$$\Phi \sim sign(U(-1,1)) \times 1$$

We can also follow Achlioptas (2003) and use the sparse random projection

$$\Phi_{ij} = \begin{cases} -\sqrt{3} & \text{, with probability} \quad 1/6 \\ 0 & \text{, with probability} \quad 2/3 \\ \sqrt{3} & \text{, with probability} \quad 1/6 \end{cases} .$$

In this paper we adopt the scheme

$$\begin{array}{l} \Pr\left(\Phi_{ij} = \frac{1}{\sqrt{\varphi}}\right) = \varphi^2 \\ \Pr\left(\Phi_{ij} = 0\right) = 2\left(1-\varphi\right)\varphi \\ \Pr\left(\Phi_{ij} = -\frac{1}{\sqrt{\varphi}}\right) = \left(1-\varphi\right)^2 \end{array} , \qquad (4)$$

where $\varphi \sim (0,1)$ (or for computational stability $\varphi \sim (0.1, 0.7)$)

Estimation of compressed VAR

We are interested in large systems, **conditionally on a known (generated)** Φ

$$y_t = B^{'}\left(\Phi X_t\right) + \varepsilon_t = B^{'}\tilde{X}_t + \varepsilon_t \tag{5}$$

so we use analytical Bayesian results instead of MCMC. In particular we define the standard natural conjugate prior:

$$\begin{aligned} vec\left(B\right) &\sim N\left(\underline{\beta}, \Sigma \otimes \underline{V}\right) \\ \Sigma &\sim IW\left(\underline{\nu}, \underline{\Sigma}\right) \end{aligned}$$

and posterior location and scale parameters for B, Σ are available analytically.

→ I won't elaborate on this, this is a standard setup for large dimensions; see Koop and Korobilis (2010), Banbura et al (2010, JAE), Giannone et al (2015, RESTAT)

Uncertainty about random projection

- Φ is not estimated from data, and we also don't know its dimensions
- In practice we generate many random $\Phi^{(r)}$, $r = 1, ..., R$ of different dimensions, then use BMA

First, we compute model r BIC as

$$BIC^{(r)} = \ln\left(\left|\overline{\Sigma}^{(r)}\right|\right) + \frac{\ln(T)}{T}m \tag{6}$$

Then posterior model probability is defined as

$$\Pr\left(M^{(r)}|y\right) \approx \frac{\exp\left(-\frac{1}{2}\Omega^{(r)}\right)}{\sum_{\varsigma=1}^{R}\exp\left(-\frac{1}{2}\Omega^{(\varsigma)}\right)}, \tag{7}$$

where $\Omega^{(r)} = BIC^{(r)} - \min BIC$ and $\min BIC$ is the lowest of the marginal likelihoods; see Kapetanios et al. (2008, JBES).

Prediction

- Our final aim is forecasting.
- In that respect, similarly to the issue of factor identification in PCA or likelihood-based analysis (e.g. Lopes and West, 2004, Statistica Sinica), we don't care about optimizing on the projection or recovering the "original" large VAR coefficients
- We also care about multi-step predictive densities (12 horizons): computationally intensive to use predictive simulation
- Instead, we work with posterior means (modes), and forecast using

$$\begin{array}{c} E\left(\mathbf{y}_{t+h}\right) = \sum_{i=0}^{h-1} \mathbf{B}^i \mathbf{c} + \mathbf{B}^h y_{t-1} \\ var\left(\mathbf{y}_{t+h}\right) = \sum_{i=0}^{h-1} \mathbf{B}^i \Sigma \left(\mathbf{B}^i\right)' \end{array} \tag{8}$$

  where all matrices here are in bold because they refer to writing the VAR in typical companion (VAR(1)) form (in cases with more than one lag); see Lutkepohl (2005).

Extension of the basic framework: Compressing the VAR covariance matrix

- In a large VAR the autoregressive coefficients can be hundreds of thousand
- The covariance matrix can also be large and have several thousand elements
- So a reasonable extension would be to compress Σ
- In order to achieve this, we specify a "triangular/structural VAR" or SVAR
- Consider the Choleski decomposition of Σ

$$\Sigma = D^{-1}H(HD^{-1})' \tag{9}$$

  where D is a lower unitriangular matrix of covariances, and H a diagonal matrix of stdevs

Extension of the basic framework: Compressing the VAR covariance matrix

The VAR can be transformed as

$$\begin{aligned} y_t &= A^{'}X_t + \varepsilon_t \Rightarrow && (10) \\ y_t &= A^{'}X_t + D^{-1}Hu_t \Rightarrow && (11) \\ Dy_t &= DA^{'}X_t + Hu_t \Rightarrow && (12) \\ (I+\tilde{D})y_t &= DA^{'}X_t + Hu_t \Rightarrow && (13) \\ y_t &= \tilde{A}^{'}X_t - \tilde{D}y_t + Hu_t && (14) \end{aligned}$$

where D has been decomposed into the identiy matrix, and a lower triangular matrix D with zeros on the diagonal.

- We now have the covariances of the original VAR as regressors, and we can compress $\tilde{D}$
- y_t in the RHS, but system estimated eq-by-eq ($\tilde{D}$ triangular)

Empirics

- We collect a “Stock-and-Watson” type dataset, i.e. 130 monthly macroeconomic variables from FRED
- We use the ones that Michael McCracken has collected in database called “FRED-MD”
- https://research.stlouisfed.org/econ/mccracken/fred-databases/
- Series stationary using same transformation codes as McCracken and Ng (2015)
- Final sample is 1960M1 - 2014M12
- We are interested in forecasting employment (PAYEMS), inflation (CPIAUCSL), and interest rate (FEDFUNDS)

VAR sizes

- We have four sets of VARs: a SMALL, a MEDIUM, a LARGE, and a HUGE
- SMALL VAR has only the three variables of interest (employment, interest, inflation)
- MEDIUM VAR has 19 variables
- LARGE VAR has 48 variables
- HUGE VAR has 119 variables
- All VARs can give forecasts of variables of interest, but imply different information sets
- **Important note: I will focus in this presentation on MEDIUM & LARGE VAR results (work is in progress)**

Evaluation of forecasts

- Initial estimation based on 50% of the sample
- We forecast $h = 1$ to 12 months ahead
- Then add one observation at the end of the sample, and repeat until obs $T - h$
- We evaluate forecasts using mean squared forecast error (MSFE) and average (log) predictive likelihoods (APLs)
- These are averages over the squared forecast errors and PLs over the last 50% of the sample
- Competing methods are BVAR with Minnesota prior exactly as in Banbura et al (2010, JAE), and FAVAR using PCA as in Bernanke et al (2005, QJE) with selection of lags and factors using BIC
- Reduced Rank VAR is in **Carriero** + co-authors? Not yet incorporated, we need MCMC which can be costly

Relative MSFE results, MEDIUM VAR

Employment

	BVARMINN	BCVAR	BCTRVAR
h=1	1.03	1.11	1.19
h=3	0.92	1.13	1.20
h=6	1.04	1.04	1.08
h=12	1.09	1.01	1.03

Inflation

	BVARMINN	BCVAR	BCTRVAR
h=1	0.97	1.08	1.08
h=3	1.05	0.99	1.00
h=6	0.99	0.99	0.98
h=12	0.99	1.00	1.00

Interest Rate

	BVARMINN	BCVAR	BCTRVAR
h=1	1.89	0.74	0.70
h=3	1.65	0.94	0.92
h=6	1.27	1.13	1.10
h=12	1.07	1.01	1.00

Relative MSFE results, LARGE VAR

Employment

	BVARMINN	BCVAR	BCTRVAR
h=1	1.47	1.19	1.28
h=3	1.10	1.01	1.10
h=6	1.06	0.93	0.99
h=12	1.05	0.91	1.01

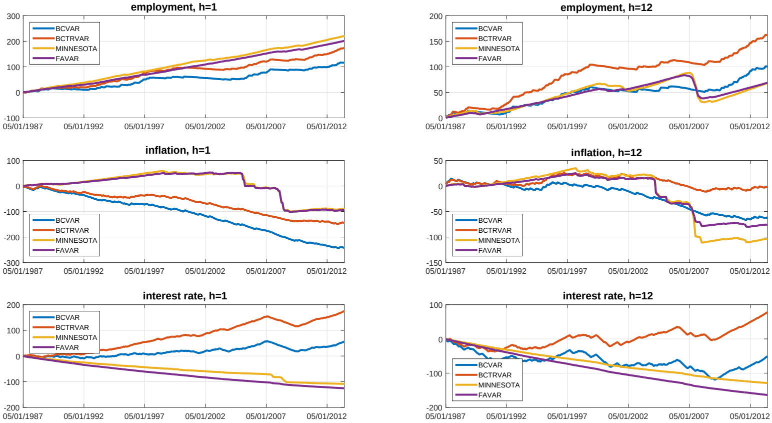
Inflation

	BVARMINN	BCVAR	BCTRVAR
h=1	0.96	1.05	1.06
h=3	1.02	0.96	0.99
h=6	1.01	0.95	0.97
h=12	0.99	0.96	0.98

Interest Rate

	BVARMINN	BCVAR	BCTRVAR
h=1	1.71	0.81	0.87
h=3	1.34	0.84	1.08
h=6	1.10	1.02	1.34
h=12	1.00	0.93	1.05

Cumulative Sum of log PLs



Comments

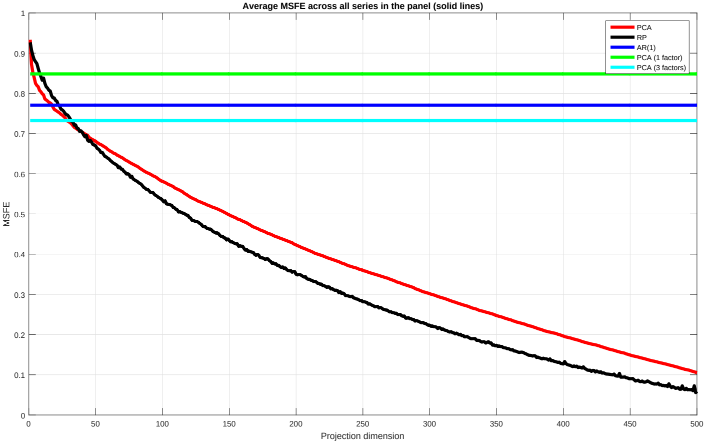
- Results look promising, especially on the predictive density side
- Compressed VAR computationally much faster than Minnesota, but less efficient than FAVAR with PCA+OLS
- For computational (stability) issues we have forecasted with differenced data
- It is in our “to do” list to examine data in log-levels (as in Banbura et al, 2010, JAE)
- After all for log level data the Minnesota/sum of coefficients prior would be much harder to beat
- We have also attempted to leave intercepts and AR(1) coeffs “un-compressed”, but no impact in forecasting
- Interesting to examine alternative compressed VAR formulations, e.g. FAVAR using RPs

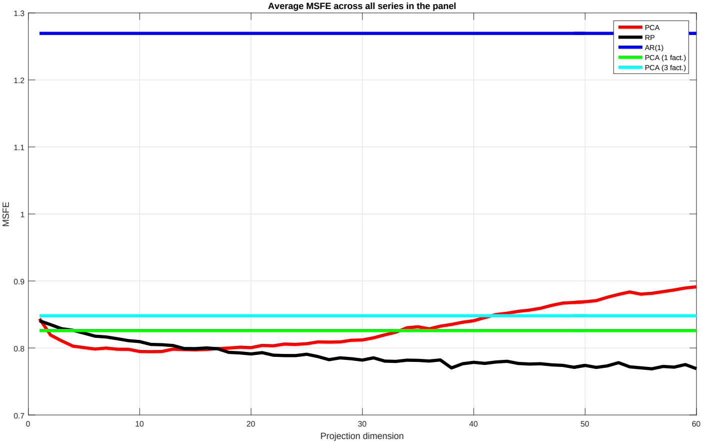
Experiments with Random Projections

- Consider the 119 macro variables (will explain those later) and define AR models of the form

$$x_{i,t} = \phi_1 x_{-i,t-1} + ... + \phi_{12} x_{-i,t-12} + \varepsilon \tag{15}$$

  where $x_{-i,t}$ denotes the vector x_t with its i-th column removed.
- Thus, the RHS matrix $z_t = (x^{'}_{-i,t-1}, ..., x^{'}_{-i,t-12})$ has $12 \times 118 = 1416$ elements
- I project $w = zR$, and use w as RHS, where R is replaced by RP, or PCA (reduced-rank regression case)
- compare with AR(2) for $x_{i,t}$, as well as diffusion index forecasting using PCA (1 & 3 factors)
- Latter takes $f_t = Lx_{-i,t}$, then uses as RHS $f_{t-1}, ..., f_{t-12}$
- Compare in-sample and out-of-sample MSE as a function of the projection dimension


Average MSFE across all series in the panel (solid lines)
PCA
RP
AR(1)
PCA (1 factor)
PCA (3 factors)
MSFE
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
50
100
150
200
250
300
350
400
450
500
Projection dimension


Average MSFE across all series in the panel
MSFE
Projection dimension
1.3
1.2
1.1
1
0.9
0.8
0.7
0
10
20
30
40
50
60
PCA
RP
AR(1)
PCA (1 fact.)
PCA (3 fact.)

Thank You!