Macroeconometrics

Lecture 10 Forecasting with Large Bayesian VARs

Tomasz Woźniak

Department of Economics
University of Melbourne

Large Bayesian VARs

Sampling from the posterior density

Sampling from predictive density

Feasible computations

Forecasting Australian real output and inflation using fat data

References:

Panagiotelis, Athanasopoulos, Hyndman, Jiang, Vahid (2019) Macroeconomic forecasting for Australia using a large number of predictors, International Journal of Forecasting

Bańbura, Giannone, Reichlin (2010) Large Bayesian Vector Auto Regressions, Journal of Applied Econometrics

Materials:

R files `L10 mcxs-N2.R` and `L10 mcxs-N117.R` for the reproduction of the results

Data file `ausmacrodata-2016.csv`

Objectives.

- To introduce challenges of working with fat data
- To present Bayesian solutions to overparemeterised models
- To forecast output and prices using 117 variables

Learning outcomes.

- Understanding some computational challenges of working with large data
- Forecasting with Bayesian VARs
- Verifying the computational time of alternative routines

Large Bayesian VARs

Bayesian VARs

Posterior density.

$$p\left(A, \Sigma | Y, X\right) = p(A|Y, X, \Sigma) p\left(\Sigma | Y, X\right)$$

$$p(A|Y, X, \Sigma) = \mathcal{MN}_{K \times N}\left(\overline{A}, \Sigma, \overline{V}\right)$$
$$p(\Sigma|Y, X) = \mathcal{IW}_N\left(\overline{S}, \overline{\nu}\right)$$

$$\overline{V} = \left(X'X + \underline{V}^{-1}\right)^{-1}$$
$$\overline{A} = \overline{V}\left(X'Y + \underline{V}^{-1}\underline{A}\right)$$
$$\overline{\nu} = T + \underline{\nu}$$
$$\overline{S} = \underline{S} + Y'Y + \underline{A}'\underline{V}^{-1}\underline{A} - \overline{A}'\overline{V}^{-1}\overline{A}$$

Large Bayesian VARs

Fat data problem.

Large Bayesian VARs are defined by the infeasibility of the OLS estimation. The problem arises when the number of variables N is large compared to the length of time series T, that is, when $1 + pN > T$

The infeasibility of the OLS estimation comes from the reduced rank of $X'X$ which then cannot be inverted.

Macroeconomic forecasting.

Consider a system of monthly macro-aggregates for monetary policy in the U.S. The data are available from 1959, which gives $T \approx 750$. Consider VAR(12). In such a case, solving $K = 1 + pN < T$ gives $N < 63$.

However, more than a hundred relevant variables, potentially useful for forecasting, is included in panels of data.

Large Bayesian VARs

Large Bayesian VARs are feasible because it is not $X'X$ to be inverted, but rather matrix:

$$X'X + \underline{V}^{-1}$$

where $\underline{V}^{-1}$ is a positive definite matrix.

Useful result.
A sum of a positive definite matrix and a singular matrix gives a positive definite matrix.

Forecasting.
Many variables may be used for forecasting with Bayesian VARs.

Large Bayesian VARs: Minnesota prior

Let the prior mean assume a random walk process:

$$\underline{A} = \left[\mathbf{0}_{N\times 1} \quad I_N \quad \mathbf{0}_{N\times(p-1)N} \right]'$$

Posterior mean of matrix A is:

$$\begin{aligned}\overline{A} &= \overline{V}\left(X'Y + \underline{V}^{-1}\underline{A}\right)\\ &= \overline{V}\left(X'X\hat{A} + \underline{V}^{-1}\underline{A}\right)\\ &= \overline{V}X'X\hat{A} + \overline{V}\,\underline{V}^{-1}\underline{A}\end{aligned}$$

a linear combination of the MLE $\hat{A}$ and the prior mean $\underline{A}$

Large Bayesian VARs: Minnesota prior

Reduced rank of $X'X$ problem.
Reduced rank of $X'X$ means that there is not sufficient information in data to inform the estimation of all of the parameters of A matrix. This matrix is not fully identified.

The feasibility of Bayesian estimation comes from additional identification information coming from prior distribution.

Forecasting using Minnesota prior.
As long as the information from data is sufficient we predict with an estimated model with parameter estimates $\hat{A}$.

Whenever the data is not informative about the parameters we predict with a random walk model with parameters $\underline{A}$

Sampling from the posterior density

Sampling from multivariate normal distribution

Let an N-vector X follow normal distribution. To draw

$$X \sim \mathcal{N}_N(\mu, \Sigma)$$

Sample independently N draws from a standard normal distribution $x_n \sim \mathcal{N}(0, 1)$ and create vector $\tilde{X} = (x_1, \ldots, x_N)$

Compute $S = \text{chol}(\Sigma)$ a Cholesky decomposition of Σ such that S is lower-triangular and $\Sigma = SS'$

Return $\mu + S\tilde{X}$ as a draw from $\mathcal{N}_N(\mu, \Sigma)$

In R you might use `rmvnorm` function from package `mvrnorm`

Sampling from matrix-variate normal distribution

Let a $K \times N$ matrix X follow a matrix-variate normal distribution. To draw

$$X \sim \mathcal{MN}_{K\times N}(M, Q, P)$$

Sample independently KN draws from a standard normal distribution $x_{k.n} \sim \mathcal{N}(0, 1)$ and create $K \times N$ matrix $\tilde{X}$ collecting the draws

Compute $L = \text{chol}(Q)$ and $C = \text{chol}(P)$ such that $Q = LL'$ and $P = CC'$

Return $M + C\tilde{X}L'$ as a draw from $\mathcal{MN}_{K\times N}(M, Q, P)$

For small K and N you might use a simple R code:

```
matrix(rmvnorm(1, mean=as.vector(M), sigma=Q%x%P), ncol=N)
```

Sampling from inverse Wishart distribution

Let an $N \times N$ positive definite matrix X follow an inverse Wishart distribution. To draw

$$X \sim \mathcal{IW}_N(S, \nu)$$

Compute $L = \text{chol}(S)$ such that $S = LL'$

Create $N \times N$ lower-triangular matrix Q by

setting its diagonal elements to $q_{nn} = \sqrt{c_{nn}}$ where $c_{nn} \sim \chi^2_{\nu-n+1}$ for $n = 1, \ldots, N$

setting its elements under the main diagonal to $q_{mn} \sim \mathcal{N}(0, 1)$ for $m > n$

Return $LQ^{-1\prime}Q^{-1}L'$ as a draw from $\mathcal{IW}_N(S, \nu)$

Sampling from inverse Wishart distribution

$$X \sim \mathcal{IW}_N(S, \nu)$$

For small N you might use an R function `rWishart` as follows

```
solve(rWishart(1, df=nu, Sigma=solve(S))[,,1])
```

Sampling from normal-inverse Wishart distribution

To sample S random draws from the distribution

$$p(A|Y, X, \Sigma) = \mathcal{MN}_{K\times N}\left(\overline{A}, \Sigma, \overline{V}\right)$$
$$p(\Sigma|Y, X) = \mathcal{IW}_N\left(\overline{S}, \overline{\nu}\right)$$

Sample S independent draws from inverse Wishart distribution:

```
Sigma.inv.posterior = rWishart(S,
        df=nu.bar,
        Sigma=solve(S.bar))
Sigma.posterior = apply(Sigma.inv.posterior, 3, solve)
```

For each draw of Σ sample a draw of A

```
A.posterior = array(NA,c(K,N,S))
for (s in 1:S){
      A.posterior[,,s] = rmvnorm(1,
         mean=as.vector(A.bar),
         sigma=Sigma.posterior[,,s]%x%V.bar)
      }
```

Sampling from predictive density

Predictive density: Bayesian approach

Joint predictive density.

$$p\left(Y_{t+h}\middle|Y_t\right) = \int p\left(Y_{t+h}\middle|Y_t, A, \Sigma\right) p\left(A, \Sigma | Y, X\right) d(A, \Sigma)$$

$$p\left(Y_{t+h}\middle|Y_t, Y, X, A, \Sigma\right) = \mathcal{N}_{hN}\left(Y_{t+h|t}(A), \mathbb{V}\text{ar}\left[Y_{t+h|t}\middle|A, \Sigma\right]\right)$$
$$p\left(A, \Sigma | Y, X\right) = \mathcal{NIW}_{K\times N}\left(\overline{A}, \overline{V}, \overline{S}, \overline{\nu}\right)$$

Predictive density: Bayesian approach

Joint predictive density.
Ignore the conditioning on Y, X, A, Σ in the notation

$$
\begin{aligned}
p\left(Y_{t+h}\middle|Y_t\right) &= p\left(\left(y_{t+h}, y_{t+h-1}, \ldots, y_{t+2}, y_{t+1}\right)\middle|Y_t\right) \\
&= p\left(y_{t+h}\middle|y_{t+h-1}, \ldots, y_{t+1}, Y_t\right) \ldots p\left(y_{t+2}\middle|y_{t+1}, Y_t\right) p\left(y_{t+1}\middle|Y_t\right)
\end{aligned}
$$

where the densities on the right-hand side are

$$
p\left(y_{t+i}\middle|y_{t+i-1}, \ldots, y_{t+1}, Y_t\right) = \mathcal{N}_N\left(\mu_0 + A_1 y_{t+i-1} + \cdots + A_p y_{t+i-p-1}, \Sigma\right)
$$

The decomposition above suggests an iterative structure of the algorithm for sampling from the joint predictive density

Predictive density: Bayesian approach

Sampling from the joint predictive density (Algorithm 2).

Sample draws from $p(A, \Sigma | Y, X)$ and

Obtain $\left\{A^{(s)}, \Sigma^{(s)}\right\}_{s=1}^{S}$

For each draw of parameters draw from the predictive density

Sample $y_{t+1}^{(s)} \sim \mathcal{N}_N\left(\mu_0^{(s)} + A_1^{(s)} y_t + \cdots + A_p^{(s)} y_{t-p}, \Sigma^{(s)}\right)$

Sample

$$y_{t+2}^{(s)} \sim \mathcal{N}_N\left(\mu_0^{(s)} + A_1^{(s)} y_{t+1}^{(s)} + \cdots + A_p^{(s)} y_{t-p+1}, \Sigma^{(s)}\right)$$

$\vdots$

Sample

$$y_{t+h}^{(s)} \sim \mathcal{N}_N\left(\mu_0^{(s)} + A_1^{(s)} y_{t+h-1}^{(s)} + \cdots + A_p^{(s)} y_{t-p+h}^{(s)}, \Sigma^{(s)}\right)$$

Obtain $\left\{y_{t+1}^{(s)}, \ldots, y_{t+h}^{(s)}\right\}_{s=1}^{S}$

Feasible computations

Large Bayesian VARs: feasible estimation

Inverting a matrix.
Computer algorithms perform $\mathcal{O}\left(N^3\right)$ to invert an $N \times N$ matrix

The Kroneckers.
To invert the covariance matrix of a matrix-variate normal posterior distribution apply

$$\left(\Sigma \otimes \overline{V}\right)^{-1} = \Sigma^{-1} \otimes \overline{V}^{-1}$$

which requires $\mathcal{O}\left(N^3\right) + \mathcal{O}\left(K^3\right)$ operations which is much less than $\mathcal{O}\left((NK)^3\right)$ that would be required if the whole posterior covariance matrix of $\text{vec}(A)$ was to be inverted.

The Kroneckers.
Specify their VARs to exploit the Kronecker structure of the covariance matrix.

Large Bayesian VARs: feasible estimation

```
> library(microbenchmark)
> N       = 10
> p       = 12

> Sigma     = rWishart(1,N+2,diag(N))[,,1]
> XX        = rWishart(1,p*N+3,diag(1+p*N))[,,1]

> microbenchmark(
+   reg   = solve(kronecker(Sigma,XX)),
+   kro = kronecker(solve(Sigma),solve(XX))
+ )
Unit: milliseconds
expr       min         lq       mean    median         uq        max neval
reg 1242.10252 1255.08545 1284.60924 1266.8586 1299.67269 1520.73370   100
kro   12.01087   12.47831   17.86607   13.5652   19.76565   85.75414   100
```

On average the computations are around 72 times faster

Large Bayesian VARs: feasible estimation

Inverting a precision matrix.

$$\overline{V}^{-1} = X'X + \underline{V}^{-1}$$

Requires computation of $\det\left(\overline{V}^{-1}\right)$ which can be too small for computer's precision of saving numbers to store it in the memory.

Apply standarisation.

Step 1 Divide the precision matrix by a constant $\frac{1}{c_v}\overline{V}^{-1}$

Step 2 Invert $\left(\frac{1}{c_v}\overline{V}^{-1}\right)^{-1}$

Step 3 Compute $\overline{V} = \frac{1}{c_v}\left(\frac{1}{c_v}\overline{V}^{-1}\right)^{-1}$

Choose c_v so that the computations are feasible.
Try such values as $c_v = \text{tr}\left(\overline{V}^{-1}\right)$ or $c_v = \prod_{k=1}^{K}\left(\overline{V}^{-1}\right)_{k.k}$

Large Bayesian VARs: feasible estimation

Inverting prior covariance matrix.
Prior covariance matrix $\underline{V}$ is often specified as a diagonal matrix.

Inverting a diagonal matrix.
The inverse of a diagonal matrix is equal to a diagonal matrix with its diagonal elements set to the inverses of the diagonal elements of the matrix to be inverted.

Inverting a diagonal matrix in R.

```
V.prior.inv = diag(1/diag(V.prior))
```

Large Bayesian VARs: feasible estimation

```
> K       = 1 + p*N
> V.inv   = diag(rgamma(K,1,1))

> microbenchmark(
+   regular   = solve(V.inv),
+   diagonal  = diag(1/diag(V.inv))
+ )
Unit: microseconds
     expr     min       lq     mean   median      uq      max neval
  regular 394.341 532.6595 559.7343 555.2675 586.169 1019.535   100
 diagonal   8.691  38.7660  55.2882  59.1645  68.725  153.467   100
```

On average the computations are around 10 times faster

Large Bayesian VARs: feasible estimation

A sparse matrix is a matrix with a large fraction of zero elements. Defining a matrix as a sparse allows R to perform less operations to compute the inverse of a matrix.

Computing $\overline{A}$ applying operations on triangular matrices.
Inverting $K \times K$ matrix $\overline{V}^{-1}$ to compute $\overline{A}$ requires $\mathcal{O}\left(K^3\right)$ operations.

Inverting a triangular matrix using dedicated programs may cut down the number of operations to $\mathcal{O}\left(K\right)$

$\overline{V}^{-1}$ is not triangular, however, its Cholesky decomposition is an upper-triangular matrix.

Large Bayesian VARs: feasible estimation

Computing $\overline{A}$ applying operations on triangular matrices.

$$\overline{V}^{-1} = X'X + \underline{V}^{-1}$$

$$C = \text{Chol}\left(\overline{V}^{-1}\right) \text{ such that } \overline{V}^{-1} = C'C$$

$$\downarrow$$

$$\begin{aligned} \overline{A} &= \overline{V}\left(X'Y + \underline{V}^{-1}\underline{A}\right) \\ &= C^{-1}C^{-1\prime}\left(X'Y + \underline{V}^{-1}\underline{A}\right) \end{aligned}$$

The algorithm computes:

Step 1: $\tilde{A} = C^{-1\prime}\left(X'Y + \underline{V}^{-1}\underline{A}\right)$ by forward substitution

Step 2: $\overline{A} = C^{-1}\tilde{A}$ by backward substitution

Large Bayesian VARs: feasible estimation

Computing $\overline{A}$ applying operations on sparse matrices in R.

```
V.bar.inv     = t(X)%*%X + V.prior.inv
C             = chol(V.bar.inv)

A.bar.tmp     = t(X)%*%Y + V.prior.inv%*%A.prior
A.tilde       = forwardsolve(t(C), A.bar.tmp)
A.bar         = backsolve(C, A.tilde)
```

Large Bayesian VARs: feasible estimation

```
> A.bar.tmp     = as.matrix(rnorm(K))
> V.bar.inv     = XX + diag(1/diag(V.inv))

> dedicated     = function(A.bar.tmp,V.bar.inv){
+   C = chol(V.bar.inv);
+   return(backsolve(C, forwardsolve(t(C), A.bar.tmp)))
+ }

> microbenchmark(
+   regular   = solve(V.bar.inv) %*% A.bar.tmp,
+   dedicated = dedicated(A.bar.tmp,V.bar.inv)
+ )
Unit: microseconds
      expr      min       lq      mean    median        uq       max neval
   regular 1253.798 1334.677 1933.1417 1433.9000 1622.8055 17622.979   100
 dedicated  286.100  301.925  467.1581  388.2285  459.4685  4780.349   100
```

On average the computations are around 4 times faster

Large Bayesian VARs: feasible estimation

Useful matrix operations.

Let X be an $N \times N$ nonsingular matrix.

$$(\Sigma \otimes X)^{-1} = \Sigma^{-1} \otimes X^{-1}$$
$$\det(cX) = c^N \det(X)$$
$$(cX)^{-1} = \frac{1}{c} X^{-1}$$

Forecasting Australian real output growth and inflation using fat data

Forecasting Australian real output and inflation

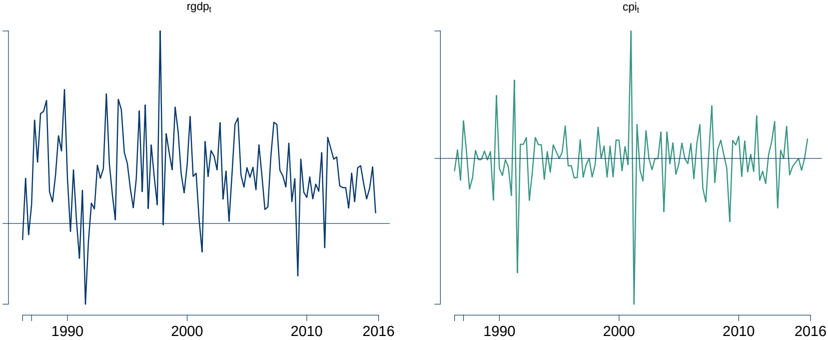
A dataset consisting of 117 quarterly macro-time series beginning in Q2 1985 was constructed by academics at Monash University and is available at http://www.ausmacrodata.org

Two related publications describe the variables and use them for forecasting Australian real output and inflation.

Information regarding dataset.
Behlul, Panagiotelis, Athanasopoulos, Hyndman, Vahid (2017) The Australian Macro Database: An Online Resource for Macroeconomic Research in Australia

Forecasting with 117 variables.
Panagiotelis, Athanasopoulos, Hyndman, Jiang, Vahid (2019) Macroeconomic forecasting for Australia using a large number of predictors, International Journal of Forecasting

Forecasting Australian real output and inflation



$$rgdp_t = \Delta RGDP_t \qquad cpi_t = \Delta^2 CPI_t$$

A dataset consisting of 117 quarterly macro-time series beginning in Q2 1985 and finishing in Q1 2016 $T = 120$ http://www.ausmacrodata.org

Forecasting Australian real output and inflation

The variables are transformed to stationary form by differentiation or log-differentiation.

Minnesota prior mean.
Therefore, the prior mean for matrix A is set to

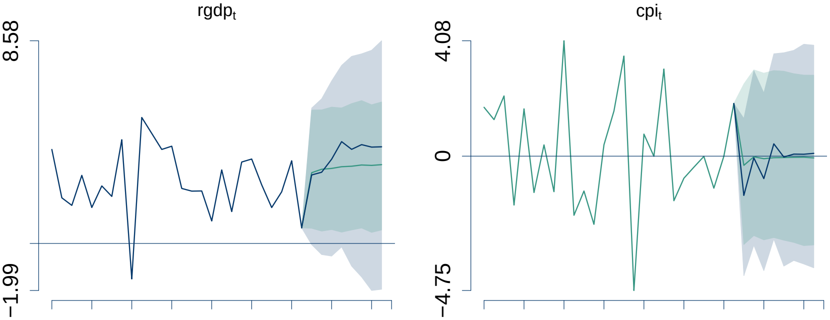
$$\underline{A} = \mathbf{0}_{K\times N}$$

and implies a white noise process $y_t = \epsilon_t$

Minnesota prior shrinkage.
The overall shrinkage parameter κ_1 is controlling the dispersion around a prior mean.

Forecasting Australian real output and inflation

Forecasting using two variables.



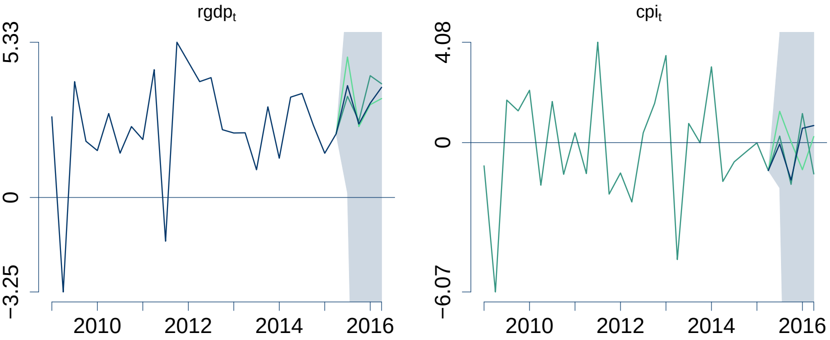
Bayesian VAR(4) with Minnesota prior and $\kappa_1 = 1$

Bayesian VAR(4) with Minnesota prior and $\kappa_1 = 0.02^2$

2-year ahead mean forecasts and 68% predictive intervals

Forecasting Australian real output and inflation

Forecasting using 117 variables.



Bayesian VAR(1) with Minnesota prior
Bayesian VAR(2) with Minnesota prior
Bayesian VAR(4) with Minnesota prior

In this model, matrices A and Σ contain jointly 61,776 unique parameters.

1-year ahead mean forecasts and 68% predictive interval for VAR(1)

Forecasting with large Bayesian VARs

Bayesian VARs are benchmark models for macroeconomic forecasting

Dedicated prior specification supports the identification and forecasting with fat data

Feasible computations thanks to application of shrinkage, Kronecker structure of covariances, and programming routines for big matrices