Numerical estimation of VAR models

P. Foschi, E.J. Kontoghiorghes
University of Neuchâtel, Switzerland

Vector Autoregressive Models

- $y_t = A_1 y_{t-1} + A_2 y_{t-2} + \cdots + A_p y_{t-p} + u_t$, where $\mathrm{E}(u_t) = 0$ and $\mathrm{Cov}(u_t, u_\tau) = \delta_{t-\tau}\Sigma$.

- For $t = 1, \ldots, m$:

$$\overset{m\times n}{\begin{pmatrix} y_m^T \\ \vdots \\ y_2^T \\ y_1^T \end{pmatrix}} = \overset{m\times np}{\begin{pmatrix} y_{m-p}^T & \cdots & y_{m-2}^T & y_{m-1}^T \\ \vdots & & \vdots & \vdots \\ y_{2-p}^T & \cdots & y_0^T & y_1^T \\ y_{1-p}^T & \cdots & y_{-1}^T & y_0^T \end{pmatrix}} \overset{mp\times n}{\begin{pmatrix} A_p^T \\ \vdots \\ A_2^T \\ A_1^T \end{pmatrix}} + \overset{m\times n}{\begin{pmatrix} u_m^T \\ \vdots \\ u_2^T \\ u_1^T \end{pmatrix}},$$

- $Y = XB + U$, where
  - $\mathrm{Vec}(U) \sim (0, \Sigma \otimes I_m)$ and
  - $(X|Y)$ is Block-Toeplitz.

OLS estimation and QR decomposition

- $\begin{pmatrix} X & Y \end{pmatrix} = \begin{pmatrix} Q_X & Q_Y & Q_N \end{pmatrix} \begin{pmatrix} R_X & R_{XY} \\ 0 & R_Y \\ 0 & 0 \end{pmatrix}$,
- $R_X \hat{B} = R_{XY}$, $\hat{U} = Q_Y R_Y$ and $\hat{\Sigma} = R_Y^T R_Y$.

Displacement structure of $(X|Y)^T(X|Y)$

- $\begin{pmatrix} X & | & Y \end{pmatrix} : m \times n(p+1)$ is Block-Toeplitz.

- $A = \begin{pmatrix} X & | & Y \end{pmatrix}^T \begin{pmatrix} X & | & Y \end{pmatrix}$ has low displacement rank.

- $\nabla_F A = A - FAF^T = G^T J G, \qquad F = Z_{p+1} \otimes I_n;$
  $J = I_{n+1} \oplus (-I_{n+1})$,

$$G = \begin{array}{cc} \begin{array}{cccc} n & n & \cdots & n \end{array} & \\ \begin{pmatrix} R_p & R_{p-1} & \cdots & R_0 \\ & y_{m-p+1}^T & \cdots & y_m^T \\ & R_{p-1} & \cdots & R_0 \\ & y_{1-p}^T & \cdots & y_0^T \end{pmatrix} & \begin{array}{c} n \\ 1 \\ n \\ 1 \end{array} \end{array},$$

where $Y_i^T = (y_{m-i}\ y_{m-i-1}\ \cdots\ y_{1-i})$, $Y_p = Q_p R_p$ and $R_i = Q_p^T Y_i$.

Displacement structure (continued)

- $A = \bar{R}_1^T \bar{R}_1 + \bar{R}_2^T \bar{R}_2 - \bar{R}_3^T \bar{R}_3 - \bar{R}_4^T \bar{R}_4$,

$$
\begin{pmatrix} \bar{R}_1 \\ \bar{R}_2 \\ \hline \bar{R}_3 \\ \bar{R}_4 \end{pmatrix}
=
\left(\begin{array}{ccccc}
R_p & R_{p-1} & R_{p-2} & \cdots & R_0 \\
 & R_p & R_{p-1} & \cdots & R_1 \\
 & & R_p & \cdots & R_2 \\
 & & & \ddots & \vdots \\
 & & & & R_p \\
 & y_{m-p+1}^T & y_{m-p+2}^T & \cdots & y_m^T \\
 & & y_{m-p+1}^T & \cdots & y_{m-1}^T \\
 & & & \ddots & \vdots \\
 & & & & y_{m-p+1}^T \\
\hline
 & R_{p-1} & R_{p-2} & \cdots & R_0 \\
 & & R_{p-1} & \cdots & R_1 \\
 & & & \ddots & \vdots \\
 & & & & R_{p-1} \\
 & y_{1-p}^T & y_{2-p}^T & \cdots & y_0^T \\
 & & y_{1-p}^T & \cdots & y_{-1}^T \\
 & & & \ddots & \vdots \\
 & & & & y_{1-p}^T
\end{array}\right)
\begin{array}{c} + \\ \\ - \end{array}
$$

- Updating: Orthogonal Transformations.

- Downdating: Hyperbolic Transformations

Adding exogenous factors (inputs)

- $y_t = Cw_t + A_1 y_{t-1} + A_2 y_{t-2} + \cdots + A_p y_{t-p} + u_t$,
  $C : n \times q, \quad w_t : q \times 1;$
- $Y = \tilde{X}\tilde{B} + U$, $\tilde{X} = (W \;\; X)$ and $\tilde{B}^T = (C^T \;\; B)$.
- $A = \left(\tilde{X} \;\; Y\right)^T \left(\tilde{X} \;\; Y\right)$ is a structured matrix.
- The generator of A with respect to $(0_{q\times q} \oplus Z_{p+1}) \otimes I_n)$ is

$$G = \begin{array}{c} \begin{array}{ccccc} q & n & n & \cdots & n \end{array} \\ \begin{pmatrix} R_W & R_{W,p} & R_{W,p-1} & \cdots & R_{W,0} \\ & R_p & R_{p-1} & \cdots & R_0 \\ & & y_{m-p+1}^T & \cdots & y_m^T \\ & R_{W,p} & R_{W,p-1} & \cdots & R_{W,0} \\ & & R_{p-1} & \cdots & R_0 \\ & & y_{1-p}^T & \cdots & y_0^T \end{pmatrix} \end{array} \begin{array}{cc} q & + \\ n & + \\ 1 & + \\ q & - \\ n & - \\ 1 & - \end{array},$$

Subset VAR models and SURE models

- $B_{ji} = 0$ for each $j \in \mathcal{S}_i$, $i = 1, \ldots, n$.
- $\mathrm{Vec}(B) = \mathrm{Vec}(S_1\beta_1 \; S_2\beta_2 \; \cdots \; S_n\beta_n) = (\oplus_i S_i)\,\mathrm{Vec}(\{\beta_i\}_n)$, where $S_i : K \times k_i$ is a selection matrix, $\beta_i : k_i \times 1$, $k_i = \#\mathcal{S}_i$ and $K = n(p+1)$.
- SURE model:

$$\mathrm{Vec}(Y) = (\oplus_i X S_i)\,\mathrm{Vec}(\{\beta_i\}) + \mathrm{Vec}(U),$$

$$\mathrm{Vec}(U) \sim (0, \Sigma \otimes I_m),$$

- Estimators: OLS, GLS, FGLS.

Model reduction (SURE models)

- $\mathrm{Vec}(Y) = (I \otimes X)(\oplus_i S_i)\,\mathrm{Vec}(\{\beta_i\}) + \mathrm{Vec}(U);$ (1)

- $\begin{pmatrix} X & Y \end{pmatrix} = \begin{pmatrix} Q_X & Q_Y & Q_N \end{pmatrix} \begin{pmatrix} R_X & R_{XY} \\ 0 & R_y \\ 0 & 0 \end{pmatrix}.$

- Multiplied by $\bar{Q} = (I \otimes Q_X \quad I \otimes Q_Y \quad I \otimes Q_N)^T$ the SURE model (1) becomes

$$\begin{pmatrix} \mathrm{Vec}(R_{XY}) \\ \mathrm{Vec}(R_Y) \\ 0 \end{pmatrix} = \begin{pmatrix} (I \otimes R)(\oplus_i S_i) \\ 0 \\ 0 \end{pmatrix} \mathrm{Vec}(\{\beta_i\}) + \begin{pmatrix} \mathrm{Vec}(V_X) \\ \mathrm{Vec}(V_Y) \\ \mathrm{Vec}(V_N) \end{pmatrix},$$

where $V_X = Q_X^T U$, $V_Y = Q_Y^T U$ and $V_N = Q_N^T U$.

- $\mathrm{Cov}(\mathrm{Vec}(V_X)) = (I \otimes Q_X^T)\,\mathrm{Cov}(\mathrm{Vec}(U))(I \otimes Q_X) = (I \otimes \Sigma).$

Model reduction (SURE models)

- $\hat{U} = X\hat{B} - Y$, $\hat{B}$: FGLS estimator of the reduced SURE model.
- ..
- $\hat{\Sigma} = \hat{U}^T\hat{U} = \hat{U}^T(Q_X Q_X^T + Q_X Q_X^T + Q_X Q_X^T)\hat{U} = \hat{V}_X^T\hat{V}_X + \hat{V}_Y^T\hat{V}_Y.$

Displacement structure of $T^T T$

- $T = [T_{i-j}]_{ij}$. $\quad T : mp \times nq, \quad T_l : p \times q$.
- $A = T^T T$ has low displacement rank.
- $\nabla_F A = A - FAF^T = G^T J G, \qquad F = Z_p \otimes I_n$;
  $J = I_{p+q} \oplus (-I_{p+q})$,

$$
G = \begin{array}{c@{}c}
\begin{array}{cccc} q & q & \cdots & q \end{array} & \\
\begin{pmatrix}
R_0 & R_{0,1} & \cdots & R_{0,n-1} \\
 & T_{-1} & \cdots & T_{1-n} \\
 & R_{0,1} & \cdots & R_{0,n-1} \\
 & T_{m-1} & \cdots & T_{m+1-n}
\end{pmatrix} & \begin{array}{c} q \\ p \\ q \\ p \end{array}
\end{array},
$$

where $T = \begin{pmatrix} X_0 & X_1 & \cdots & X_{n-1} \end{pmatrix}$, $X_0 = Q_0 R_0$ and $R_{0,i} = Q_0^T X_i$.

Displacement structure of $T^T T$

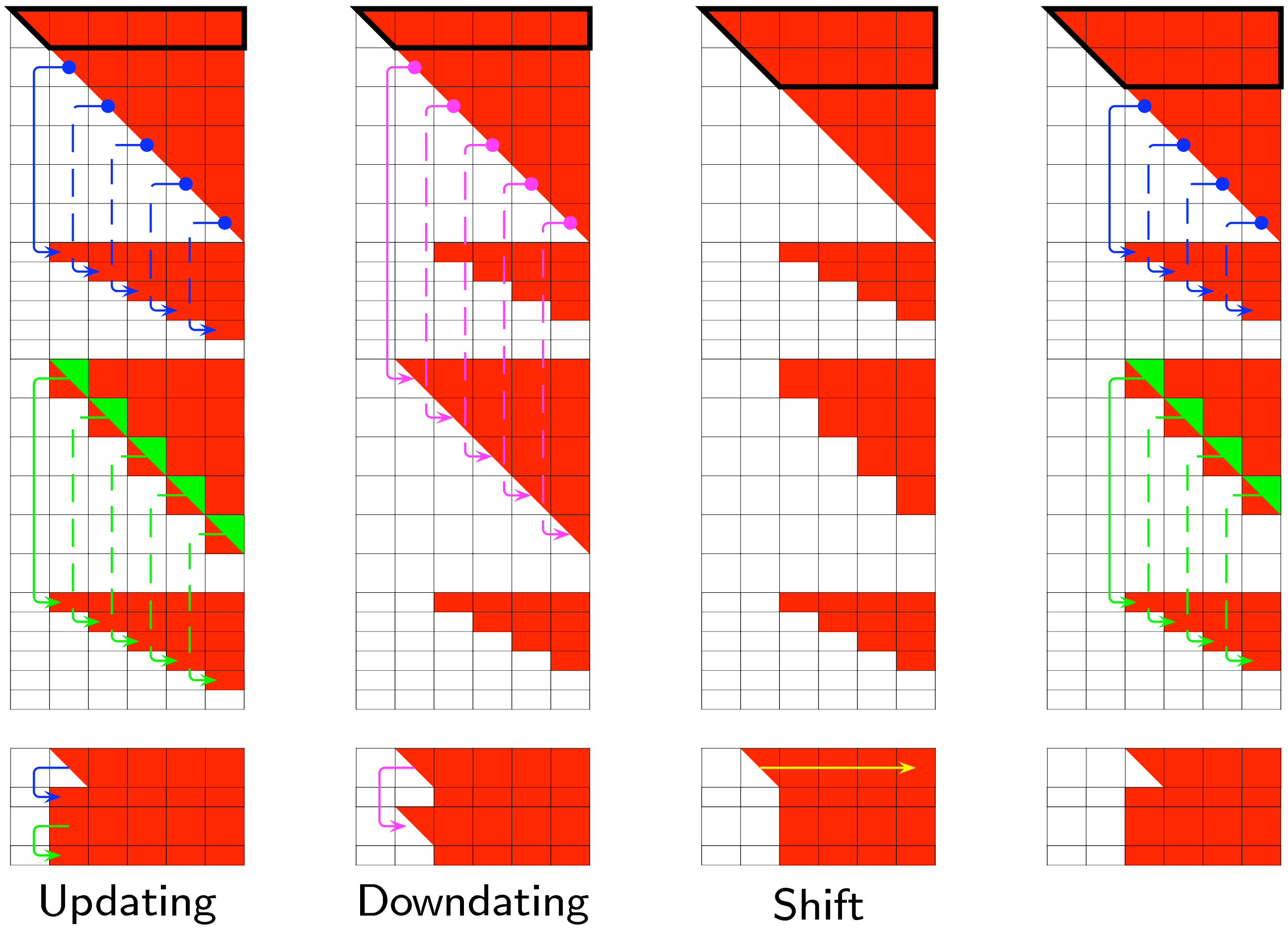
- $T^T T = R_1^T R_1 + R_2^T R_2 - R_3^T R_3 - R_4^T R_4,$

$$
\left(\begin{array}{c} R_1 \\ R_2 \\ \hline R_3 \\ R_4 \end{array}\right) = \left(\begin{array}{ccccc}
R_0 & R_{0,1} & R_{0,2} & \cdots & R_{0,n-1} \\
 & R_0 & R_{0,1} & \cdots & R_{0,n-2} \\
 & & R_0 & \cdots & R_{0,n-3} \\
 & & & \ddots & \vdots \\
 & & & & R_0 \\
 & T_{-1} & T_{-2} & \cdots & T_{1-n} \\
 & & T_{-1} & \cdots & T_{2-n} \\
 & & & \ddots & \vdots \\
 & & & & T_{-1} \\
\hline
 & R_{0,1} & R_{0,2} & \cdots & R_{0,n-1} \\
 & & R_{0,1} & \cdots & R_{0,n-2} \\
 & & & \ddots & \vdots \\
 & & & & R_{0,1} \\
 & T_{m-1} & T_{m-2} & \cdots & T_{m-n+1} \\
 & & T_{m-1} & \cdots & T_{m-n+2} \\
 & & & \ddots & \vdots \\
 & & & & T_{m-1}
\end{array}\right) \begin{array}{c} + \\ \\ - \end{array}
$$

- Updating: Orthogonal Transformations.

- Downdating: Hyperbolic Transformations (numerically unstable).

Schur algorithm



Complexity

- Generators building: $O(mn^2p)$.
- k-th step: $O(n^3k)$.
- All the $p+1$ steps: $O(n^3p^2)$.
- Total: $O(mn^2p + n^3p^2)$.
- QRD of $(X\ Y)$: $O(mn^2p^2)$.

The constant factor of the complexities is omitted. It depends on the algorithm choosen for each steps.