Drifting coefficients and volatilities

Andrea Carriero

January 2018

Dynamic Latent Variables in Macro

- We have so far considered models where all dynamics are in observables. That is, the only unobserved components are in the errors which do not exhibit relevant dynamics.
- However, in many instances, macroeconomic models involve latent dynamic variables that we wish to take into account when drawing inference.
- Some leading examples:
    1. Factor models and Factor-augmented VARs
    2. Dynamic Stochastic General Equilibrium models
    3. Time Varying Parameters models (TV-VAR, unobserved component models)
    4. Stochastic Volatilty models

Basic Set-up and Goals

- We have observed a set of N variables Y_t over T time periods.
- These in turn depends on k latent variables, or state variables, or simply "states" s_t.
- We then wish to:
  1. Estimate the parameters governing the dynamics of (Y_t, s_t).
  2. Predict and forecast both observed and latent variables.
- To reach these two goals, we need a model specifying their joint dynamics.
- This lecture: *Linear Gaussian State Space Models*, which allow for simple estimation and prediction via ML and through the use of the so-called *Kalman filter*.
- However, this class of SS is kind of restrictive - linear and Gaussian.

Linear Gaussian State Space Models (SS)

- The linear SSM's consists of two equations

$$\text{Space (measurement, observation): } Y_t = \Phi s_t + \varepsilon_t,$$

$$\text{State (transition): } s_t = F s_{t-1} + \eta_t.$$

$$\left[\begin{array}{c} \varepsilon_t \\ \eta_t \end{array}\right] \sim i.i.d.N\left(0, \left[\begin{array}{cc} \Omega_\varepsilon & 0 \\ 0 & \Omega_\eta \end{array}\right]\right).$$

- Y_t vector of N observed variables, while s_t vector of k unobserved states
- Φ and F are $N \times k$ and $k \times k$ coefficient matrices
- Intercepts or additional exogenous regressors in both equations are omitted but can be introduced easily.
- Similarly, time variation in the coefficient matrices Φ and F can be allowed

Example - ARMA as SS

- Let the observed time series Y_t solve

$$Y_t = \phi_1 Y_{t-1} + u_t + \vartheta u_{t-1}, \quad u_t \sim \text{i.i.d. } N\left(0, \sigma_u^2\right).$$

- We can rewrite this model as:

$$\begin{cases} Y_t = s_t + u_t \\ s_t = \phi_1 s_{t-1} + (\phi_1 + \vartheta) u_{t-1} \end{cases}$$

- Which is in SS form with $\Phi = 1$, $F = \phi_1$, $\varepsilon_t = u_t$, $\eta_t = (\phi_1 + \vartheta) u_{t-1}$ with $\Omega_\varepsilon = \sigma_u^2$, $\Omega_\eta = (\phi_1 + \vartheta)^2 \sigma_u^2$.
- Indeed:

$$\rightarrow s_t = \phi_1 (s_{t-1} + u_{t-1}) + \vartheta u_{t-1} = \phi_1 Y_{t-1} + \vartheta u_{t-1}$$

Example - ARMA as SS

- There are more ways to write a SS from the same model:

$$Y_t = \phi_1 Y_{t-1} + u_t + \vartheta u_{t-1}, \quad u_t \sim \text{i.i.d. } N\left(0, \sigma_u^2\right).$$

- We can write:

$$\left\{ \begin{array}{c} Y_t = s_{1t} + \vartheta s_{2t} \\ s_{1t} = \phi_1 s_{1t-1} + u_t; \\ s_{2t} = s_{1t-1} \end{array} \right.$$

- Which is in SS form with $s_t = (s_{1t}, s_{2t})'$,
$\Phi = \left[\begin{array}{cc} 1 & \vartheta \end{array} \right], \; F = \left[\begin{array}{cc} \phi_1 & 0 \\ 1 & 0 \end{array} \right], \; \varepsilon_t = 0, \; \eta_t = \left[\begin{array}{c} u_t \\ 0 \end{array} \right],$
$\Omega_\varepsilon = 0 \; , \; \Omega_\eta = \left[\begin{array}{cc} \sigma_u^2 & 0 \\ 1 & 0 \end{array} \right].$
- Indeed:

$$\begin{aligned} Y_t &= (\phi_1 s_{1t-1} + u_t) + \vartheta(\phi_1 s_{1t-2} + u_{t-1}) \\ &= \phi_1 (s_{1t-1} + \vartheta s_{1t-2}) + u_t + \vartheta u_{t-1} \end{aligned}$$

Example - ARMA as SS

- ARMA(2,1):

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + u_t + \vartheta u_{t-1}, \quad \varepsilon_t \sim \text{i.i.d. } N\left(0, \sigma_u^2\right).$$

- Can be written as:

$$\begin{aligned} Y_t &= \begin{bmatrix} 1 & \vartheta \end{bmatrix} \begin{bmatrix} s_{1t} \\ s_{2t} \end{bmatrix} = s_{1t} + \vartheta s_{2t} \\ \begin{bmatrix} s_{1t} \\ s_{2t} \end{bmatrix} &= \begin{bmatrix} \phi_1 & \phi_2 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} s_{1t-1} \\ s_{2t-1} \end{bmatrix} + \begin{bmatrix} u_t \\ 0 \end{bmatrix}, \end{aligned}$$

- Indeed:

$$\begin{aligned} Y_t &= \phi_1 s_{1t-1} + \phi_2 s_{2t-1} + u_t + \vartheta(+\phi_1 s_{1t-2} + \phi_2 s_{2t-2} + u_{t-1}) \\ &= \phi_1(s_{1t-1} + \vartheta s_{1t-2}) + \phi_2(s_{2t-1} + \vartheta s_{2t-2}) + u_t + \vartheta u_{t-1} \end{aligned}$$

Example - VARMA as SS

- Let the observed time series Y_t solve

$$Y_t = AY_{t-1} + u_t + Bu_{t-1}, \quad \varepsilon_t \sim \text{i.i.d. } N(0, \Omega_u).$$

- We can rewrite this model as

$$\begin{aligned} Y_t &= s_t + u_t, \\ s_t &= As_{t-1} + (A+B)\, u_{t-1}. \end{aligned}$$

- Indeed:

$$Y_t = A(s_{t-1} + u_{t-1}) + u_t + Bu_{t-1},$$

- In particular, $\varepsilon_t = u_t$ and $\eta_t = (A+B)\, u_{t-1}$ are uncorrelated.
- More generally, any VARMA(p,q) model with parameters θ can be formulated as a SS with $\{\Phi,\ F,\ \Omega_\varepsilon,\ \Omega_\eta\} = f(\theta)$

The Kalman Filter - Learning about states from data

- The Kalman filter is designed to produce and update linear projections of the latent variable s_t given observations of Y_t.
- Useful in its own right, and is also employed in estimation.
- The Kalman filter is a recursive algorithm that at each time point computes the current best estimate (in MSE terms) of the latent process given observations of Y_t.
- Define:

$$s_{t|s} := E\left[s_t \middle| Y_{1:s}\right], \quad P_{t|s} := \text{Var}\left[s_t \middle| Y_{1:s}\right].$$

- We then wish to do:

$$\begin{aligned} \text{Filtering} \quad : \quad & s_{t|t} \text{ and } P_{t|t}, \quad t = 1, ..., T. \\ \text{Smoothing:} \quad & s_{t|T} \text{ and } P_{t|T}, \quad t = 1, ..., T. \end{aligned}$$

Derivation of Kalman Filter

- Assume you start by knowing $s_{t-1} \sim N(s_{t-1|t-1}, P_{t-1|t-1})$. The filter is a rule to update to $s_t \sim N(s_{t|t}, P_{t|t})$ once we observe data Y_t.
- The first step is to find the joint distribution of states and data in t, conditional on past observations $(1, ..., t-1)$:

$$\left[\begin{array}{c} s_t \\ Y_t \end{array} \right] \Bigg| I_{t-1} \sim N\left(\left[\begin{array}{c} s_{t|t-1} \\ Y_{t|t-1} \end{array} \right], \left[\begin{array}{cc} P_{t|t-1} & C'_{t|t-1} \\ C_{t|t-1} & \Sigma_{t|t-1} \end{array} \right] \right) \tag{1}$$

- The moments above can be calculated easily using the equations of the system, $Y_t = \Phi s_t + \varepsilon_t, \quad s_t = F s_{t-1} + \eta_t \Rightarrow$

$$\begin{aligned}
s_{t|t-1} &= E\left[s_t | Y_{t-1}\right] = FE\left[s_{t-1} | Y_{t-1}\right] + E\left[\eta_t | Y_{t-1}\right] = Fs_{t-1|t-1}, \\
Y_{t|t-1} &= E\left[Y_t | Y_{t-1}\right] = \Phi E\left[s_t | Y_{t-1}\right] + E\left[\varepsilon_t | Y_{t-1}\right] = \Phi s_{t|t-1}. \\
P_{t|t-1} &= \text{Var}\left[s_t | Y_{t-1}\right] = FP_{t-1|t-1}F' + \Omega_\eta, \\
\Sigma_{t|t-1} &= \text{Var}\left[Y_t | Y_{t-1}\right] = \Phi P_{t|t-1}\Phi' + \Omega_\varepsilon, \\
C_{t|t-1} &= \text{Cov}\left(Y_t, s_t | Y_{t-1}\right) = \Phi P_{t|t-1}.
\end{aligned}$$

Conditional and joint normals

- We have now specified the distribution $(s_t, Y_t | Y_{t-1})$ we now look for the distribution $(s_t | Y_t, Y_{t-1}) = (s_t | Y_t)$.
- This is easy to do using basic results regarding Normal distributions: Let (a, b) be normally distributed,

$$\left[\begin{array}{c} a \\ b \end{array} \right] \sim N\left(\left[\begin{array}{c} \mu_a \\ \mu_b \end{array} \right], \left[\begin{array}{cc} \Omega_{aa} & \Omega_{ab} \\ \Omega_{ba} & \Omega_{bb} \end{array} \right] \right). \tag{2}$$

Then the conditional distribution of a conditional on b is given as

$$a|b \sim N\left(\mu_{a|b}, \Omega_{a|b}\right), \tag{3}$$

where

$$\mu_{a|b} = \mu_a + \Omega_{ab}\Omega_{bb}^{-1}\left(b - \mu_b\right), \quad \Omega_{a|b} = \Omega_{aa} - \Omega_{ab}\Omega_{bb}^{-1}\Omega_{ba}.$$

- So all is needed is to apply the result (3) when (2) is the joint distribution in (1)

Updating a linear projection

- By doing so we obtain:

$$\left[\begin{array}{c} a = s_t \\ b = Y_t \end{array}\right] \Big| \, Y_{t-1} \sim N\left(\left[\begin{array}{c} \mu_a = s_{t|t-1} \\ \mu_b = Y_{t|t-1} \end{array}\right], \left[\begin{array}{cc} \Omega_{aa} = P_{t|t-1} & \Omega_{ab} = C'_{t|t-1} \\ \Omega_{ba} = C_{t|t-1} & \Omega_{bb} = \Sigma_{t|t-1} \end{array}\right] \right)$$

  and by applying the result (3)

$$s_t | \, Y_{t-1}, Y_t \sim N\left(\mu_{a|b} = s_{t|t}, \Omega_{a|b} = P_{t|t}\right) \tag{4}$$

  with

$$s_{t|t} = s_{t|t-1} + C'_{t|t-1}\Sigma^{-1}_{t|t-1}\left(Y_t - Y_{t|t-1}\right) \tag{5}$$

$$P_{t|t} = P_{t|t-1} - C'_{t|t-1}\Sigma^{-1}_{t|t-1}C_{t|t-1} \tag{6}$$

- So we have moved from $(s_{t-1}|Y_{t-1})$ to $(s_t, Y_t|Y_{t-1})$ (prediction) and then from $(s_t, Y_t|Y_{t-1})$ to $(s_t|Y_t)$ (update).
- We can now repeat and use $(s_t|Y_t)$ to move forward to $(s_{t+1}|Y_{t+1})$

Kalman gain and updating equations

- The Kalman Filter updating equations are therefore:

$$\begin{aligned} s_{t|t} &= s_{t|t-1} + C'_{t|t-1}\Sigma^{-1}_{t|t-1}\left(Y_t - Y_{t|t-1}\right) \\ P_{t|t} &= P_{t|t-1} - C'_{t|t-1}\Sigma^{-1}_{t|t-1}C_{t|t-1} \end{aligned}$$

- Using $C_{t|t-1} = \Phi P_{t|t-1}$ (see (1)) these can be re-written as:

$$\begin{aligned} s_{t|t} &= s_{t|t-1} + K_{t|t-1}v_{t|t-1} && (7) \\ P_{t|t} &= P_{t|t-1} - K_{t|t-1}\Phi P_{t|t-1} && (8) \end{aligned}$$

with $K_{t|t-1}$

$$K_{t|t-1} = P_{t|t-1}\Phi'\Sigma^{-1}_{t|t-1} \quad (9)$$

denoting the Kalman Gain and $v_{t|t-1} = \left(Y_t - Y_{t|t-1}\right)$ denoting the 1-step ahead prediction error.

The Kalman Filter recursions

- The inputs needed in the updating equations are the moments of (1):

$$\begin{aligned}
s_{t|t-1} &= Fs_{t-1|t-1} \text{ (state prediction)} \\
v_{t|t-1} &= Y_t - \Phi s_{t|t-1} \text{ (prediction error on y)} \\
P_{t|t-1} &= FP_{t-1|t-1}F' + \Omega_\eta \quad \text{(variance of state prediction)} \\
\Sigma_{t|t-1} &= \Phi P_{t|t-1}\Phi' + \Omega_\varepsilon, \text{ (variance of prediction error)}
\end{aligned}$$

  these are called the prediction equations.
- The algorithm works as follows:
  1) Start with an initial condition $s_{t-1} \sim N(s_{t-1|t-1}, P_{t-1|t-1})$
  2) Use the 4 prediction equations above to find $s_{t|t-1}, v_{t|t-1}, P_{t|t-1}, \Sigma_{t|t-1}$,
  3) Compute the Kalman gain (9)
  4) Use the updating equations (7)-(8) to find $s_t \sim N(s_{t|t}, P_{t|t})$

Likelihood

- As a by product, the algorithm will provide the rime series of $v_{t|t-1}$ and $\Sigma_{t|t-1}$ for $t = 1, ... T$.
- So at each $t = 1, ... T$ we can compute and store:

$$l(Y_t | Y_{1:t-1}; \theta) \propto -\ln |\Sigma_{t|t-1}(\theta)| - v'_{t|t-1}(\theta) \Sigma^{-1}_{t|t-1}(\theta) v_{t|t-1}(\theta)\}$$

where

$$\theta = f^{-1}(\Phi,\ F,\ \Omega_{\varepsilon},\ \Omega_{\eta})$$

- The sum of the likelihoods of the forecast errors $\sum l_t(\theta)$ provides the likelihood of the whole system
- Therefore the KF offers a fast way to evaluate the likelihood of a SS model.

The Carter-Kohn algorithm

- A recursive algorithm to draw from the states posterior distribution
- Define the history of states and data up to time T

$$s_1, ..., s_T = \tilde{s}_T,\ y_1, ..., y_T = \tilde{y}_T,$$

  we desire to draw from $p(\tilde{s}_T|\tilde{y}_T)$.
- This posterior can be factorized as follows

$$\begin{aligned}
p(\tilde{s}_T|\tilde{y}_T) &= p(s_T|\tilde{y}_T) \times p(\tilde{s}_{T-1}|s_T, \tilde{y}_T) \\
&= p(s_T|\tilde{y}_T) \times \{p(s_{T-1}|s_T, \tilde{y}_T) \times p(\tilde{s}_{T-2}|s_{T-1}, s_T, \tilde{y}_T)\} \\
&= p(s_T|\tilde{y}_T) \times \{p(s_{T-1}|s_T, \tilde{y}_T) \times \{p(s_{T-2}|s_{T-1}, \tilde{y}_T) \\
&\quad \times p(\tilde{s}_{T-3}|s_{T-2}, s_{T-1}, s_T, \tilde{y}_T)\}\} \\
&\quad \vdots
\end{aligned}$$

Factorizing the posterior of states

- Because of the Markov property:

$$
\begin{aligned}
&= p(s_T|\tilde{y}_T) \times \{p(s_{T-1}|s_T, \tilde{y}_T) \times \{p(s_{T-2}|s_{T-1}, \tilde{y}_T) \\
&\quad \times p(\tilde{s}_{T-3}|s_{T-2}, s_{T-1}, s_T, \tilde{y}_T)\}\} \\
&\quad \vdots \\
&= p(s_T|\tilde{y}_T) \times p(s_{T-1}|s_T, \tilde{y}_{T-1}) \times p(s_{T-2}|s_{T-1}, \tilde{y}_{T-2}) \\
&\quad \times p(s_{T-3}|s_{T-2}, \tilde{y}_{T-3}) \\
&\quad \vdots \\
&= p(s_T|\tilde{y}_T) \times \prod_{t=T-1}^{1} p(s_t|s_{t+1}, \tilde{y}_t)
\end{aligned}
$$

Updating the posterior of states

- We have:
$$p(\tilde{s}_T|\tilde{y}_T) = p(s_T|\tilde{y}_T) \times \prod_{t=T-1}^{1} p(s_t|s_{t+1}, \tilde{y}_t)$$
- The last iteration of the KF gives $s_T|\tilde{y}_T$. We want to generate the terms $\prod_{t=T-1}^{1} p(s_t|s_{t+1}, \tilde{y}_t)$ and eventually obtain $p(\tilde{s}_T|\tilde{y}_T)$
- The KF gives us $s_t|\tilde{y}_t \sim N(s_{t|t}, P_{t|t})$. So the problem reduces to making the move:

$$s_t|\tilde{y}_t \sim N(s_{t|t}, P_{t|t}) \rightarrow s_t|s_{t+1}, \tilde{y}_t \sim N(s_{t|t,s_{t+1}}, P_{t|t,s_{t+1}})$$

with

$$\begin{aligned} s_{t|t,s_{t+1}} &= E[s_t|\tilde{y}_t, s_{t+1}] = E[s_t|s_{t|t}, s_{t+1}] \\ P_{t|t,s_{t+1}} &= Var[s_t|\tilde{y}_t, s_{t+1}] = Var[s_t|s_{t|t}, s_{t+1}] \end{aligned}$$

Drawing from the posterior of states

- Again this can be done using the formula for updating a linear projection
- We start with writing down the distribution of $s_t, s_{t+1}|\tilde{y}_t$:

$$\left[\begin{array}{c} a = s_t \\ b = s_{t+1} \end{array}\right] \Bigg| \tilde{y}_t \sim N\left(\left[\begin{array}{c} \mu_a = s_{t|t} \\ \mu_b = s_{t+1|t} \end{array}\right], \left[\begin{array}{cc} \Omega_{aa} = P_{t|t} & \Omega_{ab} = P_{t|t}F' \\ \Omega_{ba} = FP'_{t|t} & \Omega_{bb} = P_{t+1|t} \end{array}\right]\right)$$

where we have used:

$$COV(s_{t+1}, s_t|\tilde{y}_t) = COV(Fs_t + \eta_{t+1}, s_t|\tilde{y}_t) = FP'_{t|t}$$

Drawing from the posterior of states

- Then we use the formula for updating a linear projection:

$$a|b, \tilde{y}_t \sim N\left(\mu_{a|b}, \Omega_{a|b}\right),$$

where

$$\begin{aligned}
a &= s_t,\ b = s_{t+1} \\
\mu_{a|b} &= \mu_a + \Omega_{ab}\Omega_{bb}^{-1}\left(b - \mu_b\right) \\
&= s_{t|t} + P_{t|t}F'P_{t+1|t}^{-1}\left(s_{t+1} - s_{t+1|t}\right) = s_{t|t,s_{t+1}} \\
\Omega_{a|b} &= \Omega_{aa} - \Omega_{ab}\Omega_{bb}^{-1}\Omega_{ba} \\
&= P_{t|t} - P_{t|t}F'P_{t+1|t}^{-1}FP'_{t|t} = P_{t|t,s_{t+1}}
\end{aligned}$$

Drawing from the posterior of states

- Finally, we use the fact that $s_{t+1|t} = Fs_{t|t}$ and $P_{t+1|t} = FP_{t|t}F' + \Omega_\eta$ to get:

$$s_t | s_{t+1}, \tilde{y}_t \sim N\left(s_{t|t,s_{t+1}}, P_{t|t,s_{t+1}}\right), \tag{10}$$

with

$$\begin{aligned} s_{t|t,s_{t+1}} &= s_{t|t} + P_{t|t}F'(FP_{t|t}F' + \Omega_\eta)^{-1}\left(s_{t+1} - Fs_{t|t}\right) \\ P_{t|t,s_{t+1}} &= P_{t|t} - P_{t|t}F'(FP_{t|t}F' + \Omega_\eta)^{-1}FP'_{t|t} \end{aligned}$$

- Starting from the initial draw $s_T | \tilde{y}_T$ and the moments $s_{t|t,s_{t+1}}$ and $P_{t|t,s_{t+1}}$ can be used to recursively draw from $s_t | s_{t+1}, \tilde{y}_t \sim N(s_{t|t,s_{t+1}}, P_{t|t,s_{t+1}})$ for $t = T-1, T-2, ..., 1$.

Predetermined and exogenous variables

- The model we considered:

$$\begin{cases} Y_t = \Phi s_t + \varepsilon_t, \ \varepsilon_t \sim i.i.d.N(0, \Omega_\varepsilon), \\ s_t = F s_{t-1} + \eta_t, \ \eta_t \sim i.i.d.N(0, \Omega_\eta), \end{cases} \tag{11}$$

with ε_t and η_t independent, is more general than it seems.

- Say e.g. you want to add exogenous variables in the observation equation and an intercept in the transition equation:

$$\begin{cases} Y_t = c_Y X_t + \Phi s_t + \varepsilon_t, \ \varepsilon_t \sim i.i.d.N(0, \Omega_\varepsilon) \\ s_t = c_s + F s_{t-1} + \eta_t, \ \eta_t \sim i.i.d.N(0, \Omega_\eta) \end{cases}$$

  defining $Y_t^* = Y_t - c_Y X_t$, $s_t^* = (1, s_t)$, $F^* = [c_s', \ F']'$, and $\eta_t^* = (0', \eta_t')'$ would lead to a representation like (11)
- Watch out for identification problems!

Predetermined and exogenous variables

- Or, it might be convenient to leave the SS model with explicit intercepts and exogenous variables and just modify the Kalman Filter equations accordingly.

$$\begin{cases} Y_t = c_Y X_t + \Phi s_t + \varepsilon_t, \ \varepsilon_t \sim i.i.d.N(0, \Omega_\varepsilon) \\ s_t = c_s + F s_{t-1} + \eta_t, \ \eta_t \sim i.i.d.N(0, \Omega_\eta) \end{cases}$$

- The predictions equations involving the means will change:

$$\begin{aligned} s_{t|t-1} &= c_s + F s_{t-1|t-1} \text{ (state prediction)} && (12) \\ v_{t|t-1} &= Y_t - c_Y X_t - \Phi s_{t|t-1} \text{ (y prediction error)} && (13) \end{aligned}$$

- Note that in the example above, setting $X_t = (1, Y_{t-1}, ..., Y_{t-p})$ we have a FAVAR

Example - ARMA(2,1) again

- Consider the ARMA again:

$$Y_t = \phi_1 Y_{t-1} + u_t + \vartheta u_{t-1}, \quad u_t \sim \text{i.i.d. } N\left(0, \sigma_u^2\right).$$

$$\begin{aligned} y_t &= \phi_1 y_{1t-1} + \begin{bmatrix} 1 & \vartheta \end{bmatrix} \begin{bmatrix} s_{1t} \\ s_{2t} \end{bmatrix} \\ \begin{bmatrix} s_{1t} \\ s_{2t} \end{bmatrix} &= \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} s_{1t-1} \\ s_{2t-1} \end{bmatrix} + \begin{bmatrix} u_t \\ 0 \end{bmatrix}, \end{aligned}$$

- Note that in this case one needs to modify the filtering equations as the measurement equation contains exogenous (predetermined) variables $\phi_1 y_{1t-1}$

Time varying coefficient matrices

- Since the filter is applied for each time t it is straightforward to allow for the matrices of coefficients to be time-varying. For example:

$$\begin{cases} Y_t = \Phi_t s_t + \varepsilon_t, \ \varepsilon_t \sim i.i.d.N(0, \Omega_{\varepsilon,t}) \\ s_t = F s_{t-1} + \eta_t, \ \eta_t \sim i.i.d.N(0, \Omega_\eta) \end{cases}$$

  allows for time variation in Φ_t and $\Omega_{\varepsilon,t}$.

- The prediction equations that need to be modified are:

$$v_{t|t-1} = Y_t - \Phi_t s_{t|t-1} \text{ (y prediction error)} \quad (14)$$

$$\Sigma_{t|t-1} = \Phi_t P'_{t|t-1} \Phi'_t + \Omega_{\varepsilon,t}, \text{ (prediction error variance)} \quad (15)$$

- Setting $\Phi_t = X_t$ with X_t containing lags of Y_t gives a TVP VAR.

Applications - Time varying coefficients model

$$
\begin{aligned}
y_t &= \beta_{0t} + \beta_{1t} y_{t-1} + \beta_{2t} y_{t-2} + \ldots + \beta_{pt} y_{t-p} + e_t;\; e_t \sim iid(0, \sigma^2) \\
\beta_{it} &= \varphi_i \beta_{it-1} + v_{it};\; v_t \sim iid(0, \Omega_v),\; E[e_t, v_{is}] = 0 \;\forall i, s, t
\end{aligned}
$$

State space is:

$$
\begin{aligned}
y_t &= \underbrace{\left[\begin{array}{ccc} y_t & \ldots & y_{t-p} \end{array}\right]}_{x_t} \left[\begin{array}{c} \beta_{1t} \\ \ldots \\ \beta_{pt} \end{array}\right] + e_t \\
\left[\begin{array}{c} \beta_{0t} \\ \ldots \\ \beta_{pt} \end{array}\right] &= \left[\begin{array}{ccc} \varphi_0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \varphi_p \end{array}\right] \left[\begin{array}{c} \beta_{0t-1} \\ \ldots \\ \beta_{pt-1} \end{array}\right] + \left[\begin{array}{c} v_{0t} \\ \vdots \\ v_{pt} \end{array}\right],
\end{aligned}
$$

See Cogley and Sargent (2005) and Primiceri (2005).

Applications - Time varying coefficients model

We choose $\varphi = 1$ and specify the priors:

$$\Omega_v \sim IW(Q_0, v_0);\ \sigma^2 \sim IG(s_0/2, n_0/2)$$

the state equation is:

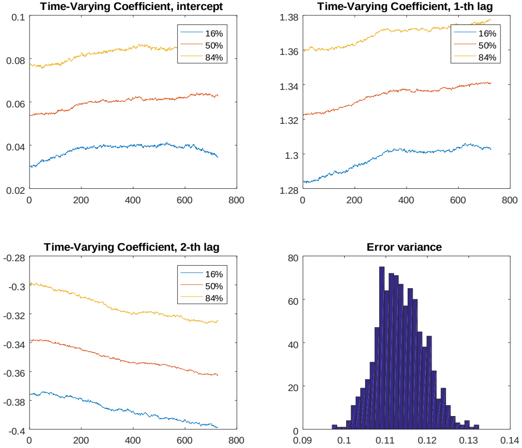
$$\beta_{it} = \beta_{it-1} + v_{it};$$

the posterior is obtained by:

1. Draw the posterior of the states using the CK algorithm: $p(\beta_{i1:T}|y, \Omega_v, \sigma^2, \beta_{i0})$
2. Draw $\Omega_v|\beta_{i1:T}, \sigma^2, y \sim IW(Q_0 + (\beta_{i2:T} - \beta_{i1:T-1})'(\beta_{i2:T} - \beta_{i1:T-1}), v_0 + T);$
3. Draw $\sigma^2|\beta_{i1:T}, \Omega_v, y \sim IG((s_0 + \sum(y_t - x_t\beta_t)^2)/2, (n_0 + T)/2)$

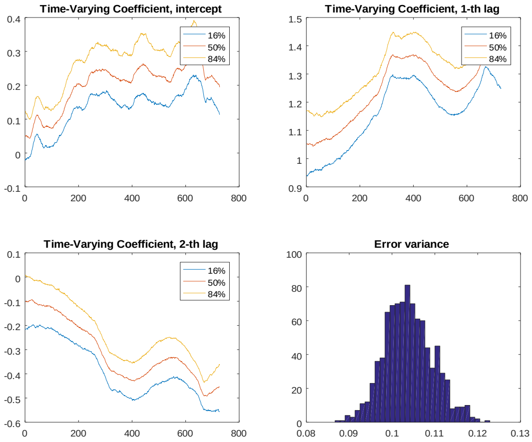
Example: ARTVP.m
Note some of the conditioning can be suppressed (in particular Ω_v and σ^2 are mutually redundant, and σ^2 is redundant in drawing the states).

Applications - Time varying coefficients model



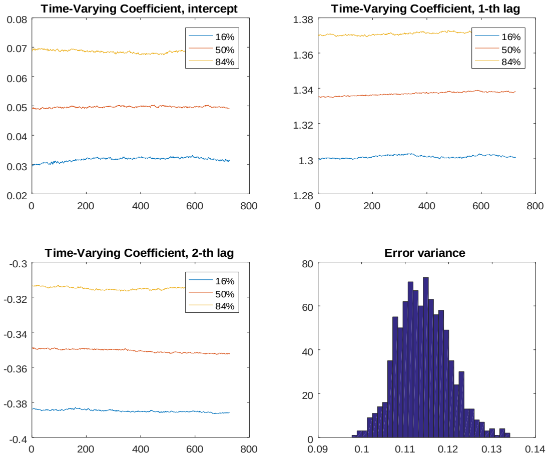
$Q_0^v = 0.01^2 \times T_0 \hat{V}_{T_0}$ where $\hat{V}_{T_0}$ is the OLS variance on a pre-sample of size T_0.

Applications - Time varying coefficients model



$$Q_0^v = 100 \times 0.01^2 \times T_0 \hat{V}_{T_0}$$

Applications - Time varying coefficients model



$$Q_0^v = \frac{1}{100} \times 0.01^2 \times T_0 \hat{V}_{T_0}$$

Applications - Stochastic Volatility models

- Model

$$\begin{aligned} Y_t &= \sqrt{h_t} z_t, \\ \log h_t &= \omega + \alpha \log h_{t-1} + \eta_t, \end{aligned}$$

  where $z_t \sim (0, 1)$.
- In particular, $h_t = E_{t-1}\left[Y_t^2\right]$ is the conditional variance of the process.
- Harvey et al. (1994) proposed to square and then take log's in the measurement equation to obtain

$$\log Y_t^2 = \kappa + \log h_t + \varepsilon_t, \quad \varepsilon_t := \log z_t^2 - \kappa, \ \kappa = E[\log z_t^2].$$

- Treating ε_t as an approximately normally distributed variable, $\left(\log Y_t^2, \log h_t\right)$ solves a linear state space model.

The Kim, Shepard and Chib (1998, KSC) algorithm

- Consider changing the error term $e_t \sim iid(0, \sigma^2)$ of the AR-TVP estimated above to:

$$e_t = \sqrt{\sigma_t^2}\varepsilon_t, \quad \varepsilon_t \sim iid(0,1),$$

  with $\ln \sigma_t^2 = \ln \sigma_{t-1}^2 + \eta_t$, that is, the error term is conditionally heteroschedastic.

- Now take the squares $e_t^2 = \sigma_t^2 \varepsilon_t^2$ and transform in logs:

$$\begin{cases} \ln e_t^2 = \ln \sigma_t^2 + \ln \varepsilon_t^2 \\ \ln \sigma_t^2 = \ln \sigma_{t-1}^2 + \eta_t \end{cases} \tag{16}$$

  which is a linear but not Gaussian state space.

- However ε_t is a Gaussian process with unit variance and hence $\ln \varepsilon_t^2$ is the log of a chi-square.

The KSC algorithm

Kim, Shepard and Chib (1998) propose to approximate the distribution of $\ln \varepsilon_t^2$ by using a mixture of normals:

$$f(\ln \varepsilon_t^2) \approx \Sigma_{i=1}^{K} q_i f_G(\ln \varepsilon_t^2 | m_i - 1.2704, v_i^2),$$

which can be written also as:

$$\begin{cases} p(s_t = i) = q_i \\ \ln \varepsilon_t^2 | s_t = i \sim N(m_i - 1.2704, v_i^2) \end{cases}$$

KSC choose K and the triplet q_i, m_i, v_i^2 that provides a good approximation:

	$s_t = 1$	$s_t = 2$	$s_t = 3$	$s_t = 4$	$s_t = 5$	$s_t = 6$	$s_t = 7$
q_i	0.0073	0.10556	0.00002	0.04395	0.34001	0.24566	0.2575
m_i	-10.12999	-3.97281	-8.56686	2.77786	0.61942	1.79518	-1.08819
v_i^2	5.79596	2.61369	5.17950	0.16735	0.64009	0.34023	1.26261

The KSC algorithm

- Under such approximation, the state space in (16) becomes tractable, conditionally on a draw of s_t
- In particular, **conditionally** on a draw of s_t, $t = 1, ..., T$ the observation equation becomes

$$\ln e_t^2 | s_t = \ln \sigma_t^2 | s_t + \ln \varepsilon_t^2 | s_t \tag{17}$$

with

$$(\ln \varepsilon_t^2 | s_t = i) \sim N(m_i - 1.2704, v_i^2) \tag{18}$$

and therefore -conditionally on s_t- the state $\ln \sigma_t^2$ can be simulated using the standard Carter-Kohn algorithm

The KSC algorithm

- Note that this means

$$(\ln e_t^2 - \ln \sigma_t^2 | s_t = i) \sim N(m_i - 1.2704, v_i^2)$$

or equivalently

$$(\ln e_t^2 | s_t = i) \sim N(\ln \sigma_t^2 + m_i - 1.2704, v_i^2)$$

with Gaussian p.d.f. $f_G(\ln e_t^2 | s_t = i)$.

- It follows that, to draw the states we can use:

$$\begin{aligned} p(s_t &= i | \ln e_t^2) \\ &\propto p(s_t = i) \times p(\ln e_t^2 | s_t = i) \\ &= q_i \times f_G(\ln e_t^2 | \ln \sigma_t^2 + m_i - 1.2704, v_i^2) \end{aligned}$$

Applications - AR with time varying variance and coefficients

We are now able to produce draws from the posterior of this -more general- model:

$$\begin{cases} y_t = \beta_{0t} + \beta_{1t} y_{t-1} + \beta_{2t} y_{t-2} + ... + \beta_{pt} y_{t-p} + \sqrt{\sigma_t^2} \varepsilon_t \\ \beta_{it} = \beta_{it-1} + v_{it}, \ i = 1, ..., N \\ \ln \sigma_t^2 = \varphi_i \ln \sigma_{t-1}^2 + \eta_t, \end{cases}$$

with:

$$\begin{aligned} \varepsilon_t &\sim iid(0,1), \ \eta_t \sim iid(0, \Omega_\eta), \ v_t \sim iid(0, \Omega_v), \\ E[\varepsilon_t, \eta_s] &= 0 \ \forall \ s, t; \ E[\eta_t, v_{i,s}] = 0 \ \forall i, s, t; \ E[\varepsilon_t, v_{i,s}] = 0 \ \forall i, s, t \end{aligned}$$

Applications - AR with time varying variance and coefficients

This model has the following parameter blocks, for which one needs to specify a prior:

$$\Omega_v \sim IW(Q_0^v, v_0^v),\ \Omega_\eta \sim IG(Q_0^\eta, v_0^\eta)$$

and the states

$$\ln \sigma_t^2,\ \beta_{it},\ i = 1, ..., N$$

To these states, we have to add the mixture states s_t necessary to be able to use the approximation:

$$\ln \varepsilon_t^2 | s_t = i \sim N(m_i - 1.2704, v_i^2)$$

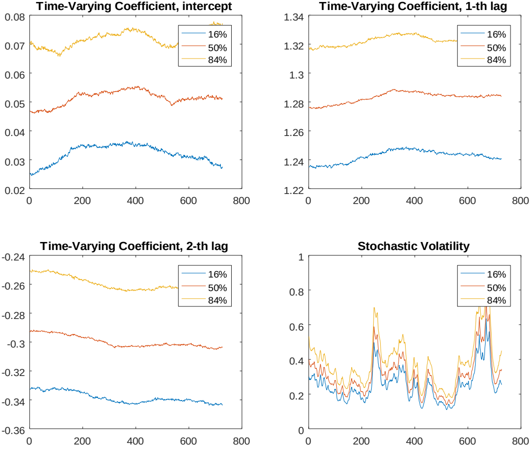
Applications - AR with time varying variance and coefficients

The algorithm draws in turn from the following distributions:

1. Draw $\Omega_v, \Omega_\eta, \beta_{i1:T}, s_{1:T} \,|\, \ln \sigma_t^2, y$
   - a Draw $\beta_{i1:T}, \Omega_v, \Omega_\eta \,|\, \ln \sigma_t^2, y$
     - i Draw $\beta_{i1:T}|y, \Omega_v, \sigma^2, \beta_{i0}$ using the CK algorithm
     - ii $\Omega_\eta|\beta_{i1:T}, \Omega_v, \ln \sigma_t^2, y \sim IG(Q_0^\eta + \Sigma(\ln \sigma_t^2 - \ln \sigma_{t-1}^2)^2, v_0^\eta + T)$
     - iii $\Omega_v|\beta_{i1:T}, \Omega_\eta, \ln \sigma_t^2, y \sim IW(Q_0^v + (\beta_{i2:T} - \beta_{i1:T-1})'(\beta_{i2:T} - \beta_{i1:T-1}), v_0^v + T)$;
   - b Draw $s_{1:T}|\beta_{i1:T}, \Omega_v, \Omega_\eta, \ln \sigma_t^2, y$
2. Draw $\ln \sigma_t^2|\Omega_\eta, \Omega_v, \beta_{i1:T}, y, s_{1:T}$ using the CK algorithm

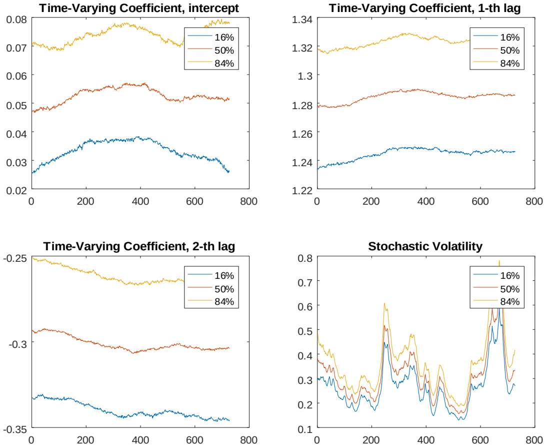
Note that step 1a and 1b are **not** intercheangeable since they constitute a draw from the joint of $p(\Omega_v, \Omega_\eta, s_{1:T}|\beta_{i1:T}, \ln \sigma_t^2, y)$

AR with time varying variance and coefficients



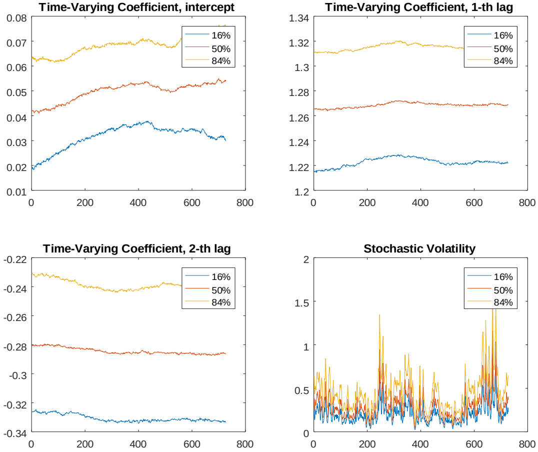
$$\frac{Q_0^{\eta}}{v_0^{\eta}-1} = 0.5, \;\; v_0^{\eta} = 3, \;\; Q_0^{\eta} = 1$$

AR with time varying variance and coefficients



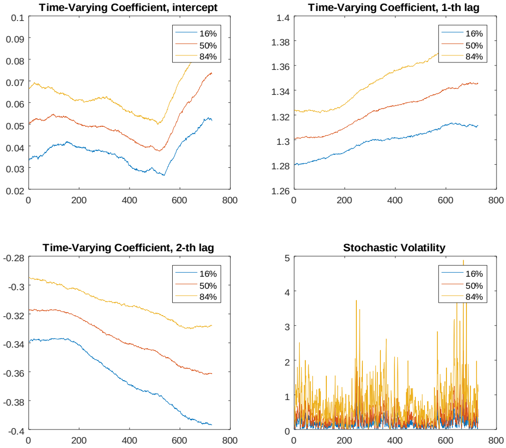
$$\frac{Q_0^\eta}{v_0^\eta - 1} = 0.005, \ v_0^\eta = 3, \ Q_0^\eta = 0.01$$

AR with time varying variance and coefficients



$$\frac{Q_0^{\eta}}{v_0^{\eta}-1} = 50, \; v_0^{\eta} = 3, \; Q_0^{\eta} = 100.$$

AR with time varying variance and coefficients



$$\frac{Q_0^{\eta}}{v_0^{\eta}-1} = 5000, \;\; v_0^{\eta} = 3, \;\; Q_0^{\eta} = 10000.$$