Using the state-space framework of JDemetra+ in R

Jean Palate
David de Antonio Liedo
Raf Baeyens

National Bank of Belgium, R&D Unit (Statistics)

Table of contents

Motivations

- JD(emetra)+
  - Java libraries for seasonal adjustment (and other time series methods)
  - Huge use of state-space models $\Longrightarrow$ Rich framework (OO design, fast algorithms . . .)
  - Main statistical references: **Durbin and Koopman**, Anderson and Moore, De Jong, Harvey, Kailath et al., Proietti . . .
- Reuse of the SSF by statisticians
  - Java too complex $\Longrightarrow$ R solution
- Existing R packages on state-spaces
  - **KFAS**, MARSS, dlm . . .
  - But . . .
    - Not always simple (definition, estimation . . .)
    - Need for a common Java/R implementation
  - $\Longrightarrow$ RJDSSF

Use cases

- Univariate models
  - Advanced SA decomposition: time varying regression, seasonal specific models
  - Temporal disggregation (mixed frequencies)
- Multivariate models
  - SUTSE models
  - Labor force survey with rotating panels (ABS, ONS)
  - Forecasting inflation (R&D + research department of NBB)

RJDSSF: Overview

- Main features
  - Automatic creation of the model (no need to define the matrices of the model)
  - Automatic identification and transformation of the parameters for ML estimation
  - Rich results
  - Common framework for a large set of (uni/multivariate) models
- Advantages
  - Power and speed of the JD+ framework
  - Most technical details are hidden
- Drawbacks
  - Designed for a special class of models
  - Limited to the pre-specified components (but extensible solution in Java)

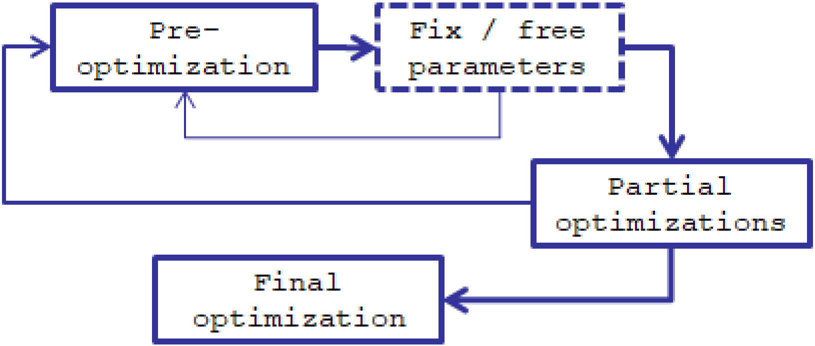
RJDSSF: Model definition

The state-space models considered in RJDSSF are defined as the concatenation of independent state blocks, on which we can apply various measurements

- State array
$$\alpha_t = \begin{pmatrix} \alpha'_{1,t} & \cdots & \alpha'_{n,t} \end{pmatrix}' \tag{1}$$
- Dynamics
$$\alpha_{i,t+1} = T_{i,t}\alpha_{i,t} + \mu_{i,t},\ \mu_{i,t} \sim N(0, [\sigma^2]V_{i,t}) \tag{2}$$
The transition matrices and the covariances of the innovations are block diagonal matrices
- Measurements
$$y_t = Z_t\alpha_t + \epsilon_t,\ \epsilon_t \sim N(0, [\sigma^2]D_t) \tag{3}$$
y_t univariate or multivariate, measurements across the blocks, independent measurement errors

RJDSSF: model estimation

- Estimation by ML
  - Transformation of the parameters $\Longrightarrow$ better properties for (quasi-)Newton methods
- Complex iterative processing (taking into account the properties of the parameters and the limits of their domain)



Example 1: STS with time varying trading days

```
# Usual BSM with time varying trading days
bsm<-function(s, seasonal="HarrisonStevens", tdgroups=c(1,2,3,4,5,6,0)){
  # create the model
  bsm<-jd3_ssf_model()
  # create the components and add them to the model
  add(bsm, jd3_ssf_locallineartrend("ll"))
  add(bsm, jd3_ssf_seasonal("s", frequency(s), type=seasonal))
  add(bsm, jd3_ssf_td("td", frequency(s), start(s), length(s), tdgroups))
  add(bsm, jd3_ssf_noise("n"))
  # create the equation
  eq<-jd3_ssf_equation("eq")
  add(eq, "ll")
  add(eq, "s")
  add(eq, "td")
  add(eq, "n")
  add(bsm, eq)
  #estimate the model
  rslt<-estimate(bsm, s, marginal=F, concentrated=T)
  return (rslt)
}
```

Example 2: Surveys with rotating panels

- Wave specific series

$$y_t^{(w)} = \boxed{Y} + b_t^{(w)} + \boxed{\varepsilon_t^{(w)}}$$

- Common process
- Wave bias
- Wave specific survey error

Y follows (for instance) a basic structural model (ARIMA modelling could also be used)

$\varepsilon_t^{(w)}$ follow a complex multi-variate auto-regressive process

Example 2: Surveys with rotating panels

- Wave specific series

$$y_t^{(w)} = \boxed{Y} + b_t^{(w)} + \boxed{\varepsilon_t^{(w)}}$$

- Common process
- Wave bias
- Wave specific survey error

Y follows (for instance) a basic structural model (ARIMA modelling could also be used)

$\varepsilon_t^{(w)}$ follow a complex multi-variate auto-regressive process

Example 2: Surveys with rotating panels

- Wave specific series

$$y_t^{(w)} = \boxed{Y} + b_t^{(w)} + \boxed{\varepsilon_t^{(w)}}$$

- Common process
- Wave bias
- Wave specific survey error

Y follows (for instance) a basic structural model (ARIMA modelling could also be used)

$\varepsilon_t^{(w)}$ follow a complex multi-variate auto-regressive process

Example 2: Surveys with rotating panels

- Wave specific series

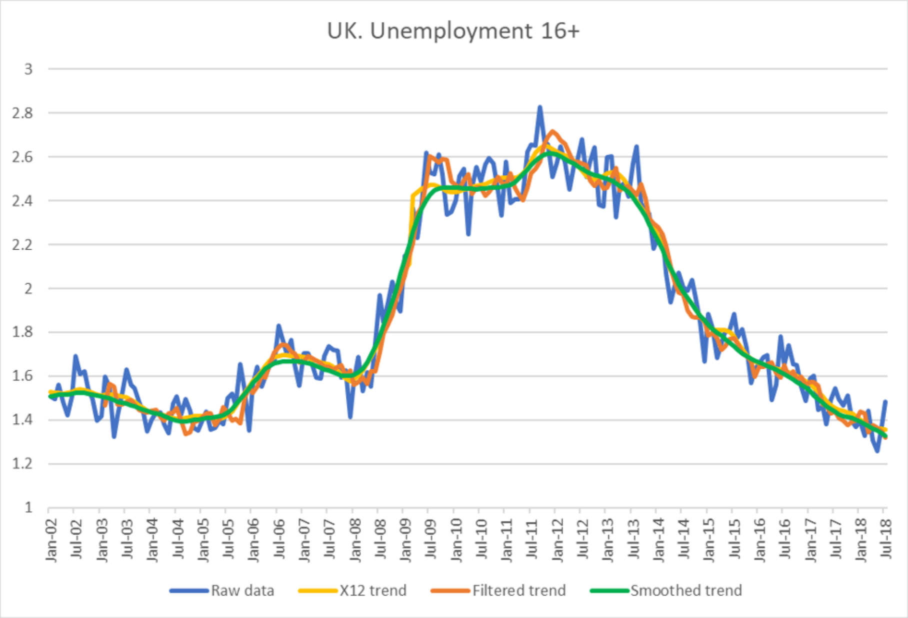
$$y_t^{(w)} = \boxed{Y} + b_t^{(w)} + \boxed{\varepsilon_t^{(w)}}$$

- Common process
- Wave bias
- Wave specific survey error

Y follows (for instance) a basic structural model (ARIMA modelling could also be used)

$\varepsilon_t^{(w)}$ follow a complex multi-variate auto-regressive process

Example 2: Labour force survey (ONS)



Example 3: Modelling inflation (UCM)

- Trend component

$$\pi_t = \boxed{\tau_t^{\pi}} + \lambda_\pi \delta_t + \zeta_\pi \vartheta_t + \eta_t^{\pi}$$

- Cyclical component
- Oil effects

$$u_t = \tau_t^{u} + \kappa_u \delta_t + \eta_t^{u}$$

$$P_t^{oil} = \tau_t^{oil} + \zeta_{oil} \vartheta_t + \eta_t^{oil}$$

Example 3: Modelling inflation (UCM)

- Trend component

$$\pi_t = \boxed{\tau_t^{\pi}} + \boxed{\lambda_\pi \delta_t} + \boxed{\zeta_\pi \vartheta_t} + \eta_t^{\pi}$$

- Cyclical component
- Oil effects

$$u_t = \tau_t^{u} + \kappa_u \delta_t + \eta_t^{u}$$

$$P_t^{oil} = \tau_t^{oil} + \zeta_{oil} \vartheta_t + \eta_t^{oil}$$

Example 3: Modelling inflation (UCM)

- Trend component
- Cyclical component
- Oil effects

$$\pi_t = \boxed{\tau_t^{\pi}} + \boxed{\lambda_\pi \delta_t} + \boxed{\zeta_\pi \vartheta_t} + \eta_t^{\pi}$$

$$u_t = \tau_t^u + \kappa_u \delta_t + \eta_t^u$$

$$P_t^{oil} = \tau_t^{oil} + \zeta_{oil} \vartheta_t + \eta_t^{oil}$$

Example 3: Modelling inflation (UCM). Cont.

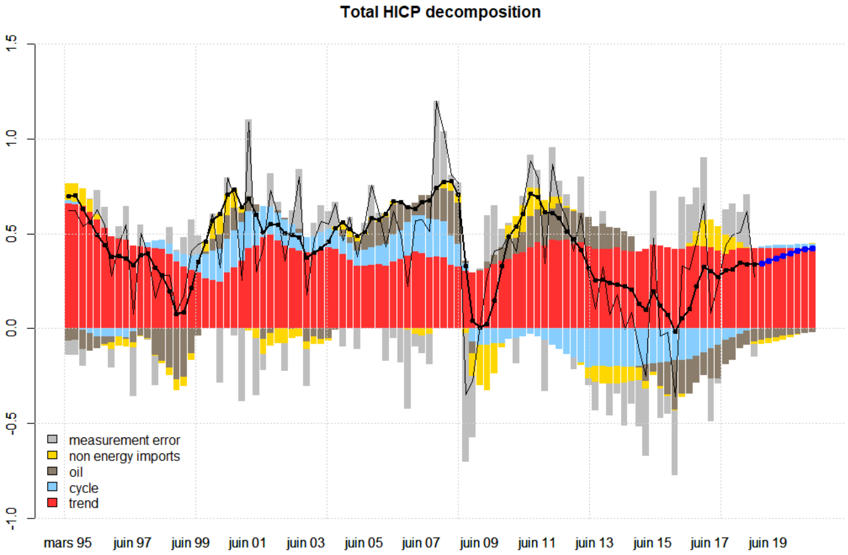
$$
\begin{aligned}
\pi_t &= \tau_t^{\pi} + \lambda_{\pi}\delta_t + \zeta_{\pi}\vartheta_t + \eta_t^{\pi} && (4)\\
\pi_t^{core} &= \tau_t^{\pi} + \lambda_{core}\delta_{t-4} + \eta_t^{core} && (5)\\
\pi_t^{m} &= \tau_t^{m} + \zeta_{m}\vartheta_t + \eta_t^{M} && (6)\\
P_t^{oil} &= \tau_t^{oil} + \zeta_{oil}\vartheta_t + \eta_t^{oil} && (7)\\
u_t &= \tau_t^{u} + \kappa_{u}\delta_t + \eta_t^{u} && (8)\\
y_t &= \tau_t^{y} + \kappa_{y}\delta_t + \eta_t^{y} && (9)\\
S_t^{Y} &= m_S + \kappa_S(\delta_t - \delta_{t-4}) + \eta_t^{S} && (10)\\
\mu_t &= m_{\mu} + \kappa_{\mu}(\delta_t - \delta_{t-4}) + \eta_t^{\mu} && (11)
\end{aligned}
$$

Surveys or variables reflecting expectations: Basselier et. al (2018), NBB

$$
\underbrace{\pi_t^{E}}_{E[\pi_{t+h}|t]} = \tau^{\pi}_{t+h|t} + \lambda_{\pi}\delta_{t+h|t} + \zeta_{\pi}\vartheta_{t+h|t} + \eta^{\pi}_{t+h|t} \qquad (12)
$$

- Qualitative survey data (e.g. Business and Consumer expectations)
- Quantitative data (e.g. ECB Survey of Professional Forecasters)

Example 2: application for euro area data: Total HICP



Future extensions

- Additional components (on demand)
  - VAR
  - ...
- Constraints
  - On parameters
  - On state items
- Regression components
- Automatic scaling of the variables
- Automatic verification of identifiability

Diffusion

- Java code (JD+ 3.0):

  https://github.com/jdemetra/jdemetra-core

- R package, examples and documentation (coming soon):

  https://github.com/nbbrd