Using the state-space framework of JDemetra+ in R

**Keywords:** State-space, JDemetra+, R.

# Introduction

State-space models provide a unified approach to a wide range of problems in the time series domain. Since its creation, the software JDemetra+, which has been officially recommended by Eurostat and the ECB for seasonal and calendar adjustment of official statistics, makes a huge use of such models. The scope of the tool will be extended in the next future to other time series topics, like temporal disaggregation, benchmarking, nowcasting and other (multi-variate) modelling techniques. That will still increase the role of state-space models in the software.

The state-space framework of JDemetra+ is based on an original object-oriented design and on advanced algorithms that make it especially powerful. We will briefly present its key features in a first point.

Even if it constitutes the kernel of many high-level routines, the state-space framework remains largely underutilized. This is basically due to the complexities of the matter and to the barrier of the programming language – Java – to access its functionalities and/or to extend them. To increase the use of the routines, we have built new modules that highly simplify in Java the creation and the estimation of univariate and multi-variate state-space models and a companion R-package for using them transparently from that well-known environment. We will expose in a second point the guiding line behind those new modules.

Finally, the new developments were driven by some very practical problems, like advanced model-based seasonal adjustment or multi-variate modelling of variables derived from surveys with complex design, like for instance the Labour Force Survey of several countries. Some results produced by the new R-package are presented in the last point.

# Methods

## State-space framework of JD+

To a large extent, the linear gaussian state-space framework of JDemetra+ corresponds to the approach of Durbin and Koopman [1] (DK hereafter). It provides the usual filtering, (disturbance) smoothing, simulation smoothing… with or without diffuse initialization, as detailed in DK. For multi-variate models, the univariate treatment of DK is provided. Beside those algorithms, the framework also proposes for non-stationary models the augmented approach of De Jong [2]. On occasion, the diffuse likelihood of DK must be replaced by the marginal likelihood, as discussed in [3]. That problem, which is often neglected, is handled in JDemetra+. Time-invariant models – even non-stationary – can be filtered by means of the fast Chandrasekhar recursions (see [4]). That algorithm improves significantly the speed of the algorithm. The framework can switch automatically to such an improvement.

State space forms may represent many different types of models. Most software will define them by means of the various matrices that describe the transition and the measurement equations, and implement the various algorithms following the matrix-based derivations that we can find in numerous papers. However, the different matrices of the system are often sparse and advanced solutions will exploit that feature to speed up some part of the processing. Nevertheless, that general approach will often lead to sub-optimal implementations, even when improvements for sparse matrices are used.

When performances are critical, it is always possible to write dedicated algorithms for specific models; that solution will always yield the most efficient code, but it cannot be considered in the context of a generic framework.

Object-oriented (OO) programming provides a third way that gives at the same time a high level of generality and of performances; JDemetra+ follows that approach. Roughly speaking, it consists in replacing the most critical matrix computations by optimized functions, which are provided by each model: the general layout of the different algorithms is common to all the models, but each model provides its optimized version of the time-consuming operations

## Modelling from R

The estimation of a parametric state space model by maximum likelihood must be split in two different problems. The first one consists in defining correctly the model (its initialization, dynamics and measurement equations) and running the suitable filtering algorithm to get the likelihood for the current parameters. The second one consists in finding, by means of some optimizer, the parameters that maximize the likelihood.

As far as R is concerned, one strategy might consist in delegating the first step to the JDemetra+ framework and using some optimizer available in R to find the maximum likelihood estimates of the parameters. That approach often leads to poor results and to degraded performances. Apart from finding good initial parameters, the main reason for explaining the convergence problems of most optimizers is linked to the structure of many state space models: their likelihood function often contains bounded parameters (variances...) as well as singularities. (Quasi-)Newton methods will usually fail when the optimum is on or very near such points. By considering the nature of the different parameters, it is however possible to improve significantly the search of the optimum (basically, we iteratively fix/relax some parameters that are near their boundaries or near singularities).

The solution we propose to get around those two problems – easy definition of models that fit the state space framework of JDemetra+ and robust optimization procedure is shortly described below.

We firstly split the state vector in independent blocks; each block has its own initialization and dynamics. Univariate models are then defined by a single equation, which reassemble the various block by means of different loading factors. Multi-variate models are straightforward extensions; they only differ by the number of equations.

The framework provides numerous building blocks – elements of the state vector or loading factors – that are used to create the actual state space model and to identify the different parameters with their properties. Examples are trend or seasonal components, regression variables with fixed or time varying coefficients, auto-regressive components, AR(I)MA components, fixed or time varying loadings, period loadings...

Once the model has been correctly defined (see below for an example), its estimation is immediate and any additional algorithms (smoothing, forecasting, simulation...) can be applied in a straightforward way.

# Results

The design of the new modules has been guided by several practical problems. One of them is the estimation of seasonal specific structural time series as defined in [5]. Such models allow the estimation of time series with systematic high volatility in different periods. The R code for generating and estimating such a model is given below (periodic volatility in January, July and August) and the noisy irregular component is presented after the code.





Another important application is the modelling of complex survey design, such as the Labor Force Survey in some countries, for instance in Australia (see [6]). The new modules allow a rather straightforward definition and estimation of multi-variate models that represent that kind of surveys. We can then derive useful information, like the measurement of the impact of changes in the survey design or more robust estimation of variables of interest.

# Conclusion

JDemetra+ contains a powerful state space framework. We have developed new modules for making the use of that framework from R, which is well known by a large audience of statisticians. Beside the integration in R, the current work has lead to significant improvements in a rather generic approach for estimating in a more robust way complex models.

# References

1. Durbin J. and Koopman S.J., "Time Series Analysis by State Space Methods", second edition. Oxford University Press (2012).
2. De Jong P. AND Chu-Chun-Lin S., "Smoothing with an Unknown Initial Condition", Journal of Time Series Analysis, 24, 2 (2003), 141-148
3. Francke, M. K., Koopman, S. J. and De Vos, A. F., “Likelihood functions for state space models with diffuse initial conditions”, Journal of Time Series Analysis, 31, 6 (2010), 407-414.
4. Kailath T., Sayed A.H. and Hassibi B., "State-Space Estimation", Prentice-Hall, Englewood Cliffs, NJ (1999).
5. Proietti T., "Seasonal specific structural time series", Studies in Nonlinear Dynamics & Econometrics, 8, 2, (2004), article 16.
6. Van den Brakel, J. and Krieg, S., “Estimation of the monthly unemployment rate through structural time series modelling in a rotating panel design”. Survey Methodology, 35, (2009) , pp. 177-190.