Automatic Forecasting of Unobserved Components Models with the UComp Toolbox for MATLAB

Pedregal DJ, Trapero JR, Villegas MA, Villegas D

UNIVERSIDAD DE CASTILLA-LA MANCHA
ETSII (CIUDAD REAL)
PREDILAB
Diego.Pedregal@uclm.es

ISF2019, Thessaloniki, Greece

Outline

1. Introduction
2. Unobserved Components models
3. Model components
4. UComp library
5. Preliminary results
6. Refined algorithm

Introduction

Automatic identification of time series models is a necessity once the big data era has come and is staying among us.

Automatic identification tools are the usual way to go in the Machine Learning area and also in some other statistical approaches, but it has never been tried out on Unobserved Components models (UC).

Base models are the Basic Structural Model of Andrew C Harvey (1989) and Durbin and Koopman (2012).

The use may be interesting either on forecasting, but also for automatic and reliable seasonal adjustment, detrending, etc.

Introduction

We are developing a library, called UComp, to implement UC models.

We are perfectly aware that making a software as general as that is very risky and demanding. But we consider this healthy in order to gain programming skills and to understand to the very roots the statistical theory behind State Space and UC models.

Unobserved Components Models

General formulation of UCs:

$$z_t = T_t + C_t + S_t + f(u_t) + I_t$$

But we restrict (initially) to:

$$z_t = T_t + S_t + I_t$$

This is the observation equation of a State Space model in which the dynamics of the components constitute the state equations. All components are gathered by block concatenation into a general State Space model.

Unobserved Components Models

Components are fully independent if the covariance matrices of all noises are block diagonal as well.

Recursive algorithms (Kalman Filter, Fixed Interval Smoother, Disturbance Smoother) provides the basis for optimal filtering, smoothing, interpolation, extrapolation, signal extraction, seasonal adjustment, detrending, etc.

They also provide the basis for Maximum Likelihood estimation.

Trend components

The identification algorithm consists of looking for the best model among a combination of possibilities for each component. Trends:

- None
- Random Walk: $Trend_{t+1} = Trend_t + \eta_t$
- Local Linear Trend:

$$\begin{bmatrix} Trend_{t+1} \\ Slope_{t+1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} Trend_t \\ Slope_t \end{bmatrix} + \begin{bmatrix} \eta_{1,t} \\ \eta_{2,t} \end{bmatrix}$$

- Damped Trend:

$$\begin{bmatrix} Trend_{t+1} \\ Slope_{t+1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & \alpha \end{bmatrix} \begin{bmatrix} Trend_t \\ Slope_t \end{bmatrix} + \begin{bmatrix} \eta_{1,t} \\ \eta_{2,t} \end{bmatrix}$$

Seasonal components

- None
- Trigonometric seasonality - Sum of block concatenated cells for a seasonal period of s samples per year, ($j = 1, 2, \ldots, s/2$ and $\omega_j = 2\pi j/s$):

  $$\begin{bmatrix} S_{j,1,t+1} \\ S_{j,2,t+1} \end{bmatrix} = \begin{bmatrix} cos\omega_j & sin\omega_j \\ -sin\omega_j & cos\omega_j \end{bmatrix} \begin{bmatrix} S_{j,1,t} \\ S_{j,2,t} \end{bmatrix} + \begin{bmatrix} \eta_{j,1,t} \\ \eta_{j,2,t} \end{bmatrix}$$

  Assuming all noise variances equal
- Same as previous but asumming variances different for each harmonic

Irregular or noise components

- None
- White noise
- ARMA model. Estimation should be carried out with care.

The full set of possible models derives from all the combinations of trends, seasonal components and noise components (36 models).

Mind that the following interesting models (some of them are not strictly speaking UC models) are particular cases of the previous ones:

- Random Walk (RW trend $+$ no seasonal $+$ no noise)
- Random Walk with drift (LLT trend with $var(\eta_{2,t}) = 0$ $+$ no seasonal $+$ no noise)
- Smooth trend (LLT trend with $var(\eta_{1,t}) = 0$ $+$ any seasonal $+$ any noise)
- Non seasonal ARMA models (no trend $+$ no seasonal $+$ ARMA noise)
- Trend $+$ no seasonal $+$ non-seasonal ARMA model

Functions

We developed a prototype in MATLAB based on SSpace (https://bitbucket.org/predilab/sspace-matlab/; Journal of Statistical Software, 2018, 87), but decided to build it in C++ plugged in R via RcppArmadillo.

Functions:

▪ sys = UCmodel(data, ...)	- Set up model
▪ sys = UCestim(sys)	- Estimate model
▪ sys = UCvalidate(sys)	- Validation
▪ sys = UCfilter(sys)	- Filtering
▪ sys = UCsmooth(sys)	- Smoothing
▪ sys = UCdisturb(sys)	- Estimating disturbances
▪ sys = UCcomponents(sys)	- Estimating components

Model set up

sys = UCmodel(data, model, h, periods, p0, cLlik, criterion, ...)

- data: time series object or vector
- model: “trend/seasonal/irregular”
  - trend: ? / none / rw / irw / llt / dt
  - seasonal : ? / none / equal / different
  - irregular: ? / none / arma(0, 0) / arma(p, q)
- h: forecast horizon
- periods: periods of seasonal component (fundamental and harmonics) for seasonal component
- p0: start values for parameters
- cLlik: concentrated out likelihood (TRUE / FALSE)
- criterion: infomation criterion for model selection

Output structure

Fields in output structure (R list):

sys$comp:	Estimated components
sys$compV:	Variance of previous
sys$p:	Estimated parameters
sys$v:	Innovations
sys$vV	Innovations variance
sys$yFit:	Fitted values
sys$yFor:	Output forecasts
sys$yForV:	Output forecasts variance
sys$a:	Estimated states
sys$P:	Estimated states variance
...	...

Features to produce robustness

Estimation is produced by Exact Maximum Likelihood. We use exact initialisation of recursive algorithms to avoid many inconvenients of the diffuse initialisation (Durbin and Koopman, 2012).

We use our own Quasi-Newton implementation of the search algorithm to have full control of the estimation process.

We concentrate one variance parameter out of the likelihood function and re-parameterise it in terms of ratios to that variance. We tend to select the higher variance to avoid the problem of dividing by zero.

Daily data on sales with weekly seasonality for the 166 products of a retailer in Spain (forecast package):

Model	**MASE**	**sMAPE**
Naive	0.740	42.149
ARIMA	0.663	37.561
ETS	0.621	37.575
Theta	**0.611**	**35.662**
UComp (?/?/?)	0.617	37.439
UC bottom line	0.525	30.47
rw/diff/arma(0,0)	0.624	37.446
rw/eq/arma(0,0)	0.629	38.308
llt/diff/arma(0,0)	0.705	41.057
none/eq/arma(0,0)	0.751	41.766

Monthly data of M3 competition (Mcomp and forecast packages):

Model	**MASE**	**sMAPE**
Naive	1.037	16.891
ARIMA	0.914	14.796
ETS	0.865	14.139
Theta	**0.858**	**13.892**
UComp (?/?/?)	0.966	14.842
UC bottom line	0.648	10.778
dt/diff/arma(0,0)	0.972	15.826
rw/eq/arma(0,0)	0.997	15.213
dt/eq/arma(0,0)	1.016	16.687
llt/eq/arma(0,0)	1.081	18.693

Algorithm steps

- Step 1: Trend test.
  Whether a trend is present in the data is decided on the basis of Mann-Kendall non-parametric test or the Augmented Dickey-Fuller unit root tests with a number of delays automatically chosen by information criteria such as Akaike's (AIC) or Schwarz's (BIC).

Algorithm steps

- Step 2: Seasonality test.
  A time series is considered to have seasonality based on a conservative and quick test. Such test assumes seasonal time series those with a one-year lag autocorrelation coefficient with a p-value smaller than 10 % (or a t-test greater than 1.645 in absolute value).
  A second sub-step here is selecting the number of harmonics on a regression of the de-trended data on sines and cosines on the fundamental frequency and its harmonics. The preliminary trend (if Step 2 indicates its presence) is calculated by a standard UC model.

Algorithm steps

- Step 3: UC model selection.
  Several models are tried out and the best is selected according to the minimization of any information criterion, either the AIC or BIC.
  The set of models to search for are a subset of all the possible combinations of components. The search may be exhaustive or restricted to a subset of the whole bunch of models depending on the results obtained in the previous steps. For example, if there is no trend, there is no need to search among models with trends and the set of models is considerably reduced.

Algorithm steps

- Step 4: ARMA model selection. A low order non-seasonal ARMA model is then selected for the innovations of the UC identified so far. ARMA models are selected according to AIC or BIC minimization, but with all ARMA models estimated in regression form using Hannan-Rissanen approximation.
  - ARMA orders should be smaller than the seasonal period to avoid confusion with the seasonal component
  - Coefficients should be estimated forcing the models to be stationary and invertible. This avoids confusion with trends (if unit roots appear in AR polynomial) and roots cancellation with trend unit roots (if unit roots appear in MA polynomial)

Algorithm steps

- Step 5: If an ARMA model is detected, then the full UC with the ARMA model embeded is jointly re-estimated by Maximum Likelihood and its AIC or BIC values computed. If the information criterion is smaller, it is retained as the best model. Otherwise the ARMA model is rejected and the best option is the UC without white noise perturbations.

Thank you for your attention!

e-mails: Diego.Pedregal@uclm.es
JuanRamon.Trapero@uclm.es

blogs: blog.uclm.es/diegopedregal/
blog.uclm.es/juanramontrapero/