

iMetrica-SSM: A State Space Modeling Graphical User-Interfaced Environment Featuring regComponent

Chris Blakely
U.S. Census Bureau

Abstract

The SSM (State Space Modeling) module for the iMetrica software is a graphical user-interfaced time series modeling and simulation environment that features computational modeling routines from the REGCMPNT software developed by the US Census Bureau. The SSM environment offers a unique time series modeling software with the primary goal of analyzing economic time series data using the most commonly used features of REGCMPNT. The environment provides an interactive interface to creating stochastic models in state space form that is both fast and boasts an intuitive design for modeling. This paper gives an in-depth overview of the module and a step-by-step guide on building and using all the modeling features of the software environment.

Keywords. SARIMA, State Space Models, Time Series, regComponent

Disclaimer This paper is released to inform interested parties of ongoing research and to encourage discussion of work in progress. The views expressed are those of the author and not necessarily those of the U.S. Census Bureau.

1 Introduction

The state-space module (SSM) in iMetrica is a versatile interface for constructing unobserved stochastic component models and regression components using the state space approach. The module features an object oriented approach to creating and stacking component models, regression components for modeling nonstochastic components, and capabilities for performing signal extraction and forecasting. Although similar to the uSimX13 module of iMetrica (see [1]) in that it is used primarily for signal extraction and forecasting of monthly economic time series, the SSM module is much more versatile due to the more general form of time series models used. For example, much more general (and multiple) SARIMA models can be specified and time dependent regression component models can also be specified. Because of this, one can propose models not available in the uSimX13 environment. Also, forecasts for individual components are also generated

automatically and are readily available. One clear advantage of the SSM comes from the versatile and wide ranging means of analyzing time series data. However, the versatility of the method makes it more challenging to use and successfully build well-specified models, making the module much less user-friendly than its co-module uSimX13. Thus the module requires much more experience in time series analysis and should be used with this warning. Through experimenting and knowing roughly what one desires to extract and analyze in time series data, the SSM environment can certainly become more user friendly, and thus we hope the SSM module can also be a sensible learning tool for time series analysis.

The SSM module uses as its computational core a Fortran program entitled REGCMPNT (short for regression-component) developed at the US Census Bureau by William Bell and others (see [5] for more details and list of other contributors). REGCMPNT is a program for Gaussian maximum likelihood (ML) estimation, signal extraction, and forecasting for univariate time series models with linear regression mean functions and error terms following ARIMA (autoregressive integrated moving average) component time series models. It uses a state space modeling approach to handle different unobserved components, and contains a variety of regression options for modeling and removing many macro-economic effects such as trading-day (both fixed and moving), moving holidays, outliers, and user-defined regressors.

The SSM module combines the computational speed and efficiency of the REGCMPNT routines for state space modeling with Kalman filtering, smoothing, and forecasting, and reassembles it into an object-oriented computational management wrapper in Java which is then controlled by users in an easy to use graphical interface. In this paper we discuss in further detail the inner workings of the SSM module, and give a few examples of how to use it to demonstrate its simplicity and robust efficiency. For the examples, we will replicate the three examples found in the original paper by Bell [4].

2 Models

We briefly outline the time series models that can be estimated by the SSM module. Consider (non)stationary observed data y_t at observation time $t = 1, \dots, N$. We attempt to model y_t using a combination of independent unobserved ARIMA components and a regression mean component in the form

$$y_t = x_t^T \beta + \sum_{j=1}^M h_{jt} \mu_t^{(j)}. \quad (1)$$

We let r be the number of regression variables and x_t denote these variables at a time point t . We then denote the corresponding vector of (fixed) regression parameters as $\beta \in \mathbb{R}^r$. The components $\mu_t^{(j)}$ for each j are independent unobserved component series following a standard ARIMA model

of the form

$$\phi_j(B)\delta_j(B)\mu_j^{(j)} = \theta_j(B)\xi_{jt}, \quad (2)$$

where $\phi_j, \delta_j(B), \theta_j(B)$ are the autoregressive, differencing, and moving-average polynomials, respectively, with the standard requirements on the zeros of ϕ_j and θ_j being on or outside of the unit circle. The ξ_{jt} are *i.i.d.* $N(0, \sigma_j^2)$ innovations with $cov(\xi_{jt}, \xi_{it}) = 0$ unless $i = j$ and $t = t'$.

To increase the modeling flexibility, h_{jt} for every j at each time observation t are fixed scalars that can be used in combination with specific choices of components μ_t for modeling Gaussian processes. However, in most modeling scenarios these scalars are simply $h_{jt} = 1$ for all t and j . We explain in further detail more on the use of h_{jt} in section 5.

3 Interface

In this section we present the overall interface and the different modeling functionalities. Unlike the other modules, this one does not automatically compute any signal extraction, forecasting, or other computational features with the direct input of data using default settings due to the larger range of modeling options as well as the sensitivity of the REGCMPNT algorithms to model specification and initial settings on parameters. Thus the user must supply all the minimal conditions for the modeling of the data in the module before any computation is done.

The interface to the SSM module is similar to all others. Principally it is broken up into four different parts: an iMetrica **menu** for data uploading and saving data and results, a SARIMA model data **simulation interface**, a **canvas** for plotting all the different components of the time series, and finally a **control board** for choosing all the different model options. Figure 1 shows the overall layout of the interface, with the four different modeling panels for selecting and/or uploading different components of the state space regComponent model. There is only one plotting canvas in this module which controls the different plots of the resulting component models.

In attempting to emulate most of the features of the REGCMPNT package in an easy to use GUI, the SSM module provides much of the nice flexibility in state space modeling while at the same time trying to streamline the options for the user to render the modeling aspect of the data as easy as possible, without compensating for loss of generality. Thus some of the more exotic printouts, regression options, and diagnostics that are available in REGCMPNT have been eliminated for the purpose of this module. We have also tried to automate as much of the modeling as possible to provide users with a quick road to signal extraction and forecasting. For example, the Kalman smoothing and forecasting for extraction of components and 36-step-ahead forecasts are computed automatically along with error variances and acfs up to lag 15 of model residuals.

One disadvantage of this module is that it requires some *a priori* knowledge of time series analysis to build effective models when using this module alone. At the present time, REGCMPNT (and consequently SSM), does not check for model misspecification or legitimacy other than residual

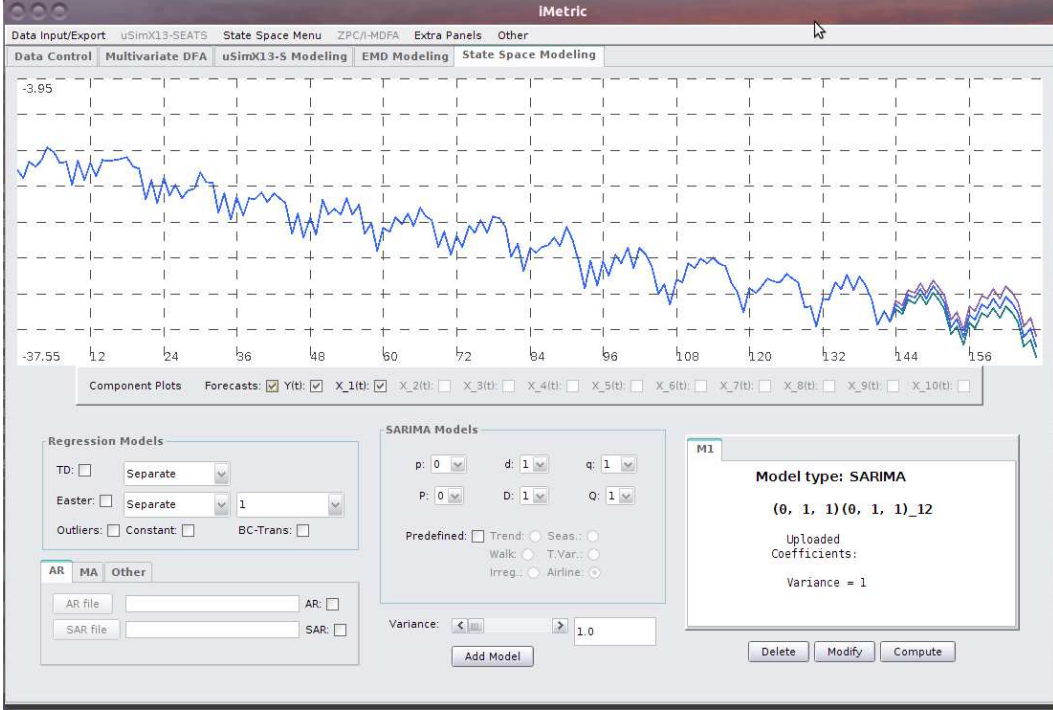


Figure 1: The overall layout of the interface with the different modeling panels and plot canvas.

ACF and histogram checks, thus a keen eye into the modeling world is required. Fortunately, with the use of hybrid-modeling in iMetrica, one can easily use SSM in conjunction with the SARIMA model to facilitate state space modeling. Please refer to Blakely [2] for an introduction to hybrid (inter-module) modeling, which is a way of combining different time series signal extraction methodologies to produce different types of extraction and forecast estimates.

The approach to state space modeling using REGCMPNT takes advantage of the robust object-oriented language of Java by symbolizing each SARIMA model structure $\phi_j(B)\delta_j(B)\mu_j^{(j)} = \theta_j(B)\xi_{jt}$ in Java as a separate object where the regression variables x_t^T are attached to the indicated model as members of the object. For example, if a trend, seasonal, and irregular model denoted by $\mu_t^{(1)} = T_t$, $\mu_t^{(2)} = S_t$, $\mu_t^{(3)} = I_t$, respectively, are constructed, three individual objects, one object representing each component model, are created. Each object for the model holds the degrees of the polynomials (the ARIMA model dimensions), the corresponding polynomial coefficients, innovation variance, and the regressor information (Easter, Trading Day, etc.) attached to that model. This allows for easy transparency and model management when building a time series model, as we will show in the Control Board section. Furthermore, easy deletion and addition of models to the expression in (1) can be managed by using Java's dynamic ArrayList data structures. We now go over the main Control board interface that manages most aspects of the SSM module.

3.1 Control Board

The control board is the collection of panels in the interface right below the plotting canvas (see Figure 1). Within the control board, the four different panels for building the models and selecting regression features, coefficients and input files are given as follows from left to right.

1. The regression panel controls the options for different regression components along with data transformation parameters. At this time, Easter, (moving) TD, fixed mean, and outliers are the only regression components available, and the Box-Cox transformation is the only data transformation option available. One can select the number of Easter days in the combo box, allocate the TD/Easter regression components to a specific model, or set them as a separate mean component. This panel should be employed after all the desired SARIMA component models have been entered in the modeling environment.
2. Directly below the regression panel, the coefficient panel enables the user to enter in manually the coefficients for each polynomial $\phi(B)$, $\theta(B)$, $\Phi(B)$, $\Theta(B)$ or upload them from a file. The format for entering them in manually is to type each one individually separated only by a space, starting with the coefficient for the first element of the associated polynomial. For example `.50 -.24` or `0.5034 -0.24143` would work for an $MA(2)$ model. However `0.50, -.24` will generate an error. Furthermore, if the dimensions of the model being constructed and those of the coefficients being entered or uploaded do not agree, the model dimensions are automatically changed to the dimensions of the data being entered. So always make sure dimensions agree. One can then fix the coefficients by checking off the checkbox respective of the parameters. If the box is not checked, the coefficients are assumed to be used as an initialization in the model estimation procedure. Alternatively, the user can fix ALL parameters by checking the box designated for "All". This will fix all polynomial coefficients along with the selected variance. Finally, the option for uploading a file for the scalars h_{jt} is available (the default is $h_{jt} \equiv 1$ for all t). This must be done *before* pressing "Add" to add the model to the environment. Once all the parameters have been entered or uploaded for the defined model, it is then safe to press "Add" button to add the model with coefficients to the model list.
3. The center panel controls the choice of the j -th SARIMA model. Notice the top portion designated for choosing the dimensions p, d, q, P, D, Q of the model. The maximum values for p, q are 11, for P and Q are 2, and for d and D are 4. The user can also select from one of the commonly used predefined models which are defined as

- Trend: $(1 - B)^2 \mu_t = (1 - \theta_\mu B) \epsilon_t, \epsilon_t \sim N(0, \sigma^2)$
- Seasonal: $\sum_{i=0}^{11} B^i \mu_t = \theta(B) \epsilon_t, \epsilon_t \sim N(0, \sigma^2)$

- Irregular: $\mu_t = \epsilon_t \sim N(0, \sigma^2)$
- Random Walk: $(1 - B)\mu_t = \epsilon_t \sim N(0, \sigma^2)$
- Time-Varying Coefficient: regression coefficients β_i following independent random walks
- Airline: $(1 - B)(1 - B^{12})\mu_t = (1 - \theta B)(1 - \Theta B)\epsilon_t, \epsilon_t \sim N(0, \sigma^2)$

Once the model has been selected, the variance can then be adjusted using the scrollbar at the bottom. The model is then added to the modeling environment to be estimated by pressing the “Add” button.

4. The modeling environment panel. This panel displays tabs on all the models that are to be used for estimation in REGCMPNT using state space modeling. When a new model is added to the environment, the properties of the model including SARIMA dimensions, variance, regression component assignments, and any AR,MA coefficients that have been assigned as fixed are displayed in an information panel. Up to 10 models (or 5 if time-varying coefficient Trading-Day is used), can be set into the modeling environment. Adjustments to the model can be made after being added to the environment by using the “Modify” button. A model can also be deleted by simply selecting the tab of the unwanted model and pressing “Delete”. Once all the desired models have been entered and the regression components have been assigned to one of the models or designated as “separate”, a press of the “Compute” button will estimate each of the component models along with the assigned regression components.

3.2 Canvas and Plots

The canvas board in the SSM offers the user the choice of plotting the different outputs when they become available after the computation of the model has finalized. In the initiation of the canvas, the only data available to be plotted is the original input time series data in its original scale. Once a model 1 has been successfully computed a series of checkboxes underneath the canvas become activated. A number of different plots that can now be plotted are estimates of the M stochastic components chosen for the computation of the model, along with their respective 24-step ahead forecasts, if available. All components are plotted on their original scale (the scale of the original time series), meaning that visualizing a nonstationary component together with a zero mean (or close to zero mean) component such as an irregular or seasonal will be most likely infeasible. Only components on the same scale should be viewed simultaneously. To view a component on a different scale from the others, one must check ‘off’ in the checkboxes panel all the components that do not share the same scale as the desired component. In other words, if more than one plot are to be viewed simultaneously, they should be on the same scale.

The forecasts of 24-steps-ahead are computed automatically in the SSM module and can be toggled on/off using the forecast checkbox. This will also plot the 24-step-ahead forecasts of the

original data, when the plot original data checkbox is selected.

To plot the signal extraction error variance for each component, clicking on the checkbox to the far right of the plot choices panel will give access to each time series. This process will simply replace the data of each component in the plot canvas.

3.3 Menu

The top menu bar contains all the options necessary for controlling data, output/input to and from files, and inter-module communication. The menu of options specific to the SSM module contain little difference from that of the SARIMA module menu. As with all the other modules, one can upload data from a file in the top menu under the “State Space” tab. The time series data file must have extension `.dat` and be column-wise with either only the time series observations, or in ‘data observation’ form, with the date of the observation followed by a space and then the time series observation. Furthermore, one can import data from a sequence of data files with the file names of all the `.dat` files contained in a `.dta` file. This is convenient if a collection of series are to be modeled using the same state space model specifications.

A feature unique to the SSM module menu comes from the ability to save not only the extracted signal estimated values, but also the model specifications. After a `regComponent` model 1 has been successfully computed, all the components are available to be printed to a file. To do this, go to the menu and click ‘Update Components’. This will produce a list of all the available components. Moving the mouse over the component number will highlight the plot of the data on the canvas if the component is selected in the checkbox panel. Clicking on the component in the list will then print the data to a file named “`iMetrica-ssm-component-number.dat`”. The format of the file is as follows (row wise):

1. Row 1: N number of observations in series
2. Rows 2 to $N+1$: N time series points
3. Row $N+2$: model dimensions (order of SARIMA model)
4. Rows $N+3$ to end : estimated model coefficients

This format enables retrieving of the signal extraction data and model in the SARIMA modeling module for further modeling of each component using canonical decompositions. For example, if one chooses to model a time series using an Airline model plus sampling error, one can select an Airline model in the model panel, and then an additional irregular process supplemented with a file denoting the standard errors at each time point given by the scalars h_t for $t = 1, \dots, N$. The coefficients are then estimated for the Airline model in the `REGCMPNT` routines and then the data along with the parameters can be saved in a file to be exported to the SARIMA module. Once in the SARIMA

module, the file can then be uploaded and further decomposed using canonical decompositions into a seasonal, irregular, and trend component. For information on this inter-module hybrid modeling, see Blakely [2].

4 Step-by-Step

The typical routine for modeling in the SSM module always begins with loading time series data into the system. The data will automatically be plotted on the canvas once stored in SSM, with horizontal hash marks and dotted lines marking every 12 data points (assuming data is monthly).

The next step is to choose the SARIMA models as in (1). To select the models, the first step will be to choose the model dimensions. One can either manually choose the SARIMA model by selecting the p, d, q, P, D, Q manually from combo boxes, or select one of the predefined SARIMA models from the panel below, which includes the models most typically used (Airline, Trend, Seasonal, etc.). Once the dimensions have been chosen, a variance, either fixed or initialized r (more on this later) is chosen using the scrollbar below the model choice box. Once content with the variance, the final step before sending it to the stack of models is to set the coefficients for the ARMA, seasonal AR (SAR), and/or seasonal MA (SMA), components in the model. The coefficients can either be set by manually typing them in the text box corresponding to the component, starting with the coefficient for the first lag of the component, all the way to the final lag, separated by a space, or they can be loaded from an input file. If manually typed in, simply press Enter on the keyboard to set the coefficients into the model. Checkboxes next to each text box indicate whether the coefficients are fixed or initiated. To include the variance as a fixed parameter, the third tabbed panel includes a checkbox for fixing all parameters of the given model.

Once the model has been selected and all parameters including the innovation variance have been set, the model can then be created as an object in Java and uploaded into the sequence of component models to be estimated in 1 by pressing the “Add Model” button. Once loaded, the model shows up in the tabbed panel to the right of the model selection. At this point it is a good idea to verify all information is correct in the model before computing. Once verified correct, additional models can be added by following the same procedure as outlined here. Up to 10 models can be entered, unless a time-varying trading day model is to be used in which case only up to 4 models can be entered (since 6 models will be dedicated to the time-varying trading day model).

The final step before computation is to add the regression components. This is done once all stochastic models are entered and coefficients have been verified as correctly entered. If any information is incorrect, the “Delete” button can be used to delete a certain model and change the information. To add the regression components, first select the checkbox associated with the regression component in the regression panel and then select which model to associate the regression component. The first entry “separate” will maintain the component as a separate effect in the data

rather than attached to an unobserved component.

The Easter regression component has an additional combo box to choose the number of days up to which the effect is computed. 29. The Easter regression component has an additional combo box to choose the "window" of the Easter regressor, that is, the level of activity changes on the wth day before the Easter holiday, and remains at the new level until the day before Easter. Furthermore, a trading day component can be implemented with time-varying coefficients or fixed. To select time-varying, check the "TV" option and then add Trading-Day model in the model selection. If 4 or less non Trading Day models have been allocated, then this will send 6 random-walk models to the model list, where each random walk represents one of the 6 coefficients in the Trading-Day regression model.

Lastly, if the log of the series must be modeled, click the checkbox for Box-Cox transformation labeled "BC-Trans". At this point in SSM, only log transformations are available. Generalizations of this transformation will perhaps be available in future versions of iMetrica.

Once all desired regression effects have been added and attributed to a model or indicated as a separate component, the model is now ready for computation. If all models have been successfully inputted and the "Compute" button has been clicked, the regCmpnt subroutines are launched and the Kalman smoothing, filtering, forecasting algorithms are iterated. If the decomposition is successful, then estimates of all components in the model are now available to be plotted on the plot canvas. To plot them, simply click the checkbox below the canvas associated with the specific model.

To illustrate this process of building and computing a model in SSM succinctly, we go through two examples that replicate the results found in Bell [5].

5 Examples

5.1 US Retail Sales of Department Stores

In the first example we look at building a structural model with a TD and Easter regression component assigned to a seasonal stochastic component plus trend and irregular noise. We consider a structural model with Easter and Trading Day regressor for data that is the logarithm of monthly retail sales of U.S. department stores from January, 1967 to December, 1993. Both TD and Easter regression effects are commonly used when modeling monthly retail sales data.

The model will have the form

$$y_t = x_t^T \beta + \gamma_t + \mu_t + \epsilon_t, \quad \epsilon_t \sim N(0, \sigma_\epsilon^2), \quad (3)$$

where

$$\begin{aligned}
(1 - B)^2 \mu_t &= (1 - \theta_\mu B) \zeta_{1,t} \\
(1 + \sum_{i=1}^{11} B^i) \gamma_t &= (1 - \theta_1 B - \dots - \theta_{10} B^{10}) \zeta_{2,t} \\
x_t^T \beta &= \sum_{i=1}^6 \beta_i T_{i,t} + \beta_E E_{10,t}.
\end{aligned} \tag{4}$$

The innovations $\zeta_{i,t}$, $i = 1, 2$ are independent white noise series and the series requires a Box-Cox (log) Transform due to growing variance over time.

With a clean interface (to make sure, one can select “Clean” from the SSM menu to erase all data from the SSM module), the first step is to load the sales data into the module producing automatically a plot of the data on the canvas. In this example, the data is in the file “tseries1.dat” where the time series observations are stored in a column. Once the data can be seen on the canvas, we begin by building each SARIMA model individually in (3) by using the central control panel and the coefficient uploader. Starting with the seasonal component γ_t , we click on “Predefined” models and then select “Seasonal” on the radio button panel. This automatically sets a SARIMA structure with the differencing operator $(1 + \sum_{i=1}^{11} B^i)$ and MA component $(1 - \theta_1 B - \dots - \theta_{10} B^{10})$. Other Seasonal components with different MA dimensions are certainly plausible and can be entered manually using the SARIMA dimension combo boxes in the central control panel, however we stick with this commonly used seasonal component form in this software for simplicity.

The next step is to then set the variance to 60 as an initialization in the Kalman filtering routines (the choice of 60 was chosen due to previous analysis of the data by other authors such as [5]). The last step needed is to set the parameters for the MA component. The state space algorithms improve in speed and robustness tremendously when one can provide the coefficients directly, or at least set them to plausible initial settings, thus this is always highly recommended. In this case, we upload the coefficients directly from a file where all ten coefficients are lined up in a column starting with the coefficient for θ_1 . We do this by clicking on “MA file” and then searching for the file in the file dialog box. Careful attention must be made that the dimensions are correct in the file, namely there must be 10 numbers in a column to correspond to each coefficient. In this example, my seasonal MA file is called “iMetrica-seas.sig”, which is the naming convention used by the SARIMA module. These MA coefficients come from a derivation of a trigonometric seasonal component used by Harvey with a single variance parameter, which was later restructured by Bell [4] into a SARIMA model, where only one parameter, namely the variance of the model was considered. We then select “Fix MA” to set these coefficients as fixed in the Kalman filtering and smoothing algorithm. Now that the model dimensions, variance, and coefficients have been addressed, it can now be created into an object and set into the modeling environment by pressing “Add” on the central control panel. We take care of the regression components after all stochastic

models have been entered.

The second model is the trend model governed by $(1 - B)^2\mu_t = (1 - \theta_\mu B)\zeta_{1,t}$. Here, we select “Trend” in the radio button panel and initialize the variance to 200. With only one coefficient in this model, we can simply manually type it into the MA coefficient box, and we initialize it to .8 and then press enter to upload it. Since there is only two parameters in this model, we allow them to be estimated through the MLE algorithm in REGCMPNT. We then add this to the modeling mix by pressing “Add”.

Finally, to model the irregular component, we select “Irreg.” on the radio button panel, and then initialize the variance at 200. Since no (AR/MA) coefficients are to be estimated, the model is complete and we press “Add”.

Now that all the necessary stochastic components are uploaded, we can now associate each of the regression components to one of the component models or select them as separate. The default as shown in the combo box in the regression component panel is “separate”, however here we associate them with the seasonal component. To model both Easter and Trading-Day, simply click the Checkbox for both, then choose which models to associate them with in the combo box next to it (in this case the seasonal component is the first in the list of models, thus we set it to “1” in the combo box for both TD and Easter).

Finally, we set the number of Easter days to consider (up to 15 is possible, and the default is 1). Lastly, since the data to be considered is the log of the sales, we must choose Box-Cox transformation “BC-Trans” in the checkbox. Not doing a log transformation might produce estimation problems in REGCMPNT so be careful.

After all desired regression components have been selected, we are ready to model the data with the set features, thus we push “Compute” in the model panel to the far right. If all features have been adequately accounted for in the data, the computation of the state space algorithms should only take a few milliseconds and automatically, all modeled components with their forecasts and error variances should be ready for viewing in the canvas and plotting panel. Figure 2 shows the SSM control panel after modeling has taken place. Notice in the plot selection box just below the canvas, three plots are available in addition to the original time series data. These correspond to the three signals that were computed in the structural model, namely the seasonal, trend, and irregular. The forecasts are also available for each individual series except the irregular component (shown in Figure 2) since irregular components do not get individual forecast treatment.

In the menu, one can click “Update Components” and this will make available all the different estimates of the stochastic components to be exported to output data files for other software or other iMetrica modules. For example, to print out the estimated seasonal component to a file named “iMetrica-ssm-seaonal.dat”, we simply click ”Update Components” to enable and update the available components in the menu, and then click “Component 1”.

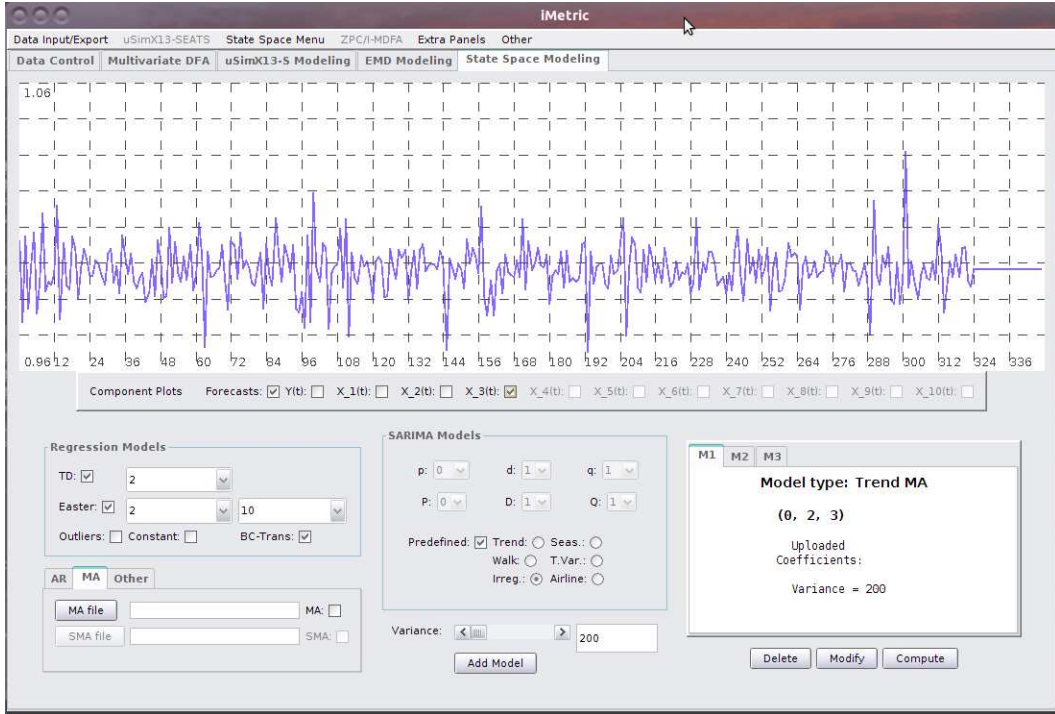


Figure 2: The overall layout of the interface for example 5.1 after the three models have been loaded into the SSM modeling queue and the computation of the model by pressing “Compute” has been accomplished.

5.2 Modeling a Time Series with Sampling Error

In this example, we demonstrate the functionality of uploading a scalar sequence h_t for a specified stochastic model. This example comes from the fourth example in Bell [5]. We attempt to build a AR(2) sampling error plus Airline model for a value of construction put in place (VIP) series. The value of construction put in place (VIP) is a U.S. Census Bureau publication measuring the value of construction installed or erected at construction sites during a given month. Estimates of the time series come from the monthly Construction Progress Reporting Survey and are augmented with estimates of other Census data, administrative records, and trade association data. Further information, including more recent data and revised historical VIP estimates, can be found at <http://www.census.gov/const/www/c30index.html>.

In Nguyen, Bell, and Gomish [6], time series modeling and signal extraction methods were employed for improving the VIP estimates. We illustrate how this can be done in the SSM module while using a sampling error component.

For this problem, we consider an Airline model for the signal component Y_t of the time series y_t and in addition an AR(2) process for the sampling error where a scalar sequence h_t representing sampling error coefficients of variation (CV) will be attached to the AR(2) process. Thus the model

is given as

$$y_t = Y_t + e_t$$

where

$$\begin{aligned} (1 - B)(1 - B^{12})Y_t &= (1 - \theta_1 B)(1 - \theta_2 B^{12})\zeta_{1,t} \\ e_t &= h_t \tilde{e}_t \\ (1 - \phi_1 B - \phi_2 B^2)\tilde{e}_t &= \zeta_{2,t}. \end{aligned} \tag{5}$$

Here, the values for the Airline model satisfy $|\theta_1| < 1$ and $|\theta_2| < 1$ and $\zeta_{i,t}$, $i = 1, 2$ are independent white noise series. We also assume the series needs a Box-Cox Transform.

In this model, the parameters ϕ_1 and ϕ_2 are estimated via the Yule-Walker equations obtained from the estimated sampling error autocorrelation and the innovation variance parameter σ_2 is then determined from the formula for the variance of an AR(2) process so that $Var(\tilde{e}_t) = 1$.

No initial values are specified for the nonseasonal and seasonal MA parameters θ_1 and θ_2 ; their initial values will thus be set to the ARMA parameter default initial value of .1. The initial value set for σ_1 by the variance argument was taken from a previous analysis of the series. The new feature in this example is the specification of the scale factors h_t , which are set to the CV values.

As before, we begin by loading the Airline model first into the modeling platform by clicking “Predefined” models and then clicking “Airline” model. The variance is set to 0.01 and then loaded into the modeling panel by pressing “Add”. The second model is chosen as an AR(2) by choosing 2 in the p combo box and then choosing the variance as .34 which was determined by *a priori* analysis so that $Var(\tilde{e}_t) = 1$. The coefficients for the two AR parameters can be entered in the text box for the AR component as .60 .246, with the two numbers only separated by a space, and then hitting “Enter” on the keyboard to load them in. With the model structure already defined, we now can proceed with attaching the CV process h_t to the AR process. This is done in the panel right below the *Regression Models* panel. Click “other” and in the box is access to a file uploader button. Click on it and the file chooser dialog box will pop up. Choose the file that has the h_t values found (this file must be in column data format, with one entry per line). Careful attention must be paid when uploading this file in that it must have the same amount of values as the number of observations in the original time series to be modeled y_t . Once the file is chosen, click the “Add” button. This will attach the values to the AR(2) in the model box. Once this is done, the AR(2) model can then be added to the queue of models by clicking the “Add Model” button. This AR(2) model should then be visible in the models queue and an h_t coefficients label should be shown in the model description. If not, delete the model from the queue and start again. Figure 3 shows what the control panel should look like once the two models are entered in the queue. Notice how the second model M2 shows that the file `h_coeff.dat` for h_t has been loaded and that the AR coefficients are fixed. This tabbed model queue is a good way of keeping track of the models.

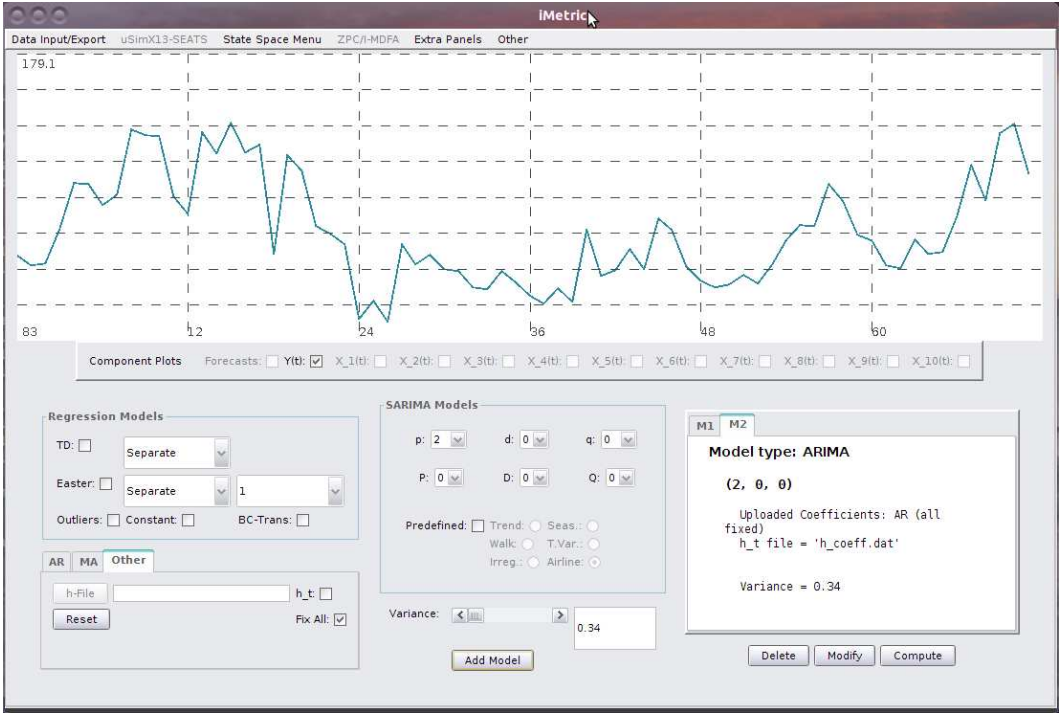


Figure 3: The overall layout of the interface for example 2 after the two models have been loaded into the SSM modeling queue.

Once both the Airline model and the $AR(2)$ model with h_t coefficients are properly defined (both models should be visible in the queue), press “Compute” right below the model queue panel. If all parameters were defined correctly, the computation phase of the iMetrica software should only take a few milliseconds, a couple seconds at most, depending on the computer’s processor speed. If it takes longer, there was most likely a model misspecification somewhere (perhaps a wrong variance defined) or the h_t file is not valid. This indicates there were modeling problems and the state space algorithms were not able to converge to an optimal solution. In this case, the model queue should be emptied and the modeling process should be restarted again. In the case the computations went smoothly, the plotting canvas should now have two additional plots available to be plotted in addition to the original time series. Furthermore, access to forecasts and the pointwise forecast standard-errors should be available.

References

[1] Blakely, C. (2012) iMetrica-uSimX13: A Graphical User-Interfaced Time Series Modeling and Simulation Environment Featuring X-13-ARIMA-SEATS

[2] Blakely, C. (2012) Hybrid Econometrics: Signal Extraction and Forecasting with iMetrica. *In*

progress

- [3] Blakely, C. (2012) iMetrica-Data Simulator: A Graphical User-Interface for the Uploading, Simulating, and Analyzing Multidimensional Time Series.
- [4] Bell, W. (2003) On RegComponent Time Series Models and Their Applications. In AC Harvey, SJ Koopman, N Shephard (eds.), *Space and Unobserved Component Models: Theory and Applications* chapter 12. Cambridge University Press, Cambridge.
- [5] Bell, W. (2011) REGCMPNT: A Fortran Program for Regression Models with ARIMA Component Errors. *Journal of Statistical Software*, Vol 41, Issue 7.
- [6] Nguyen T, Bell W, Gomish JM (2002). Investigating Model-Based Time Series Methods to Improve Estimates from Monthly Value of Construction Put-in-Place Surveys. In *Proceedings of the American Statistical Association* pp. 2470-2475.
- [7] U. S. Census Bureau (2012), *X-13ARIMA-SEATS Reference Manual, Version 1.0*, U.S. Census Bureau, U.S. Department of Commerce (<http://www.census.gov/ts/x13as/docX13AS.pdf>)