# iMetrica-uSimX13: A Graphical User-Interfaced Time Series Modeling and Simulation Environment Featuring X-13ARIMA-SEATS

Chris Blakely
U.S. Census Bureau

**Abstract**

The uSimX13 module for the iMetrica software is a graphical user-interfaced time series modeling and simulation environment that features computational modeling routines from the X-13ARIMA-SEATS (X-13A-S) software published by the US Census Bureau. The uSimX13 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 X-13ARIMA-SEATS. The environment provides a large array of classical and modern goodness-of-fit tests to assess different model fits of the data, many different graphical representations of the time series data, adaptive time series decomposition capabilities, and much more all while being accessible to both beginners in the field of econometrics wanting to visualize frequently used tools, and practitioners wanting to obtain forecasts, seasonal/trend adjustments, and/or test and apply regression components to their data. This paper gives an in-depth overview of many of the modeling features of the uSimX13 software environment.

**Keywords.** X-13ARIMA-SEATS, SARIMA, Time Series,

**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 iMetrica software project began in 2010 as a simple graphical user interface for the X-13ARIMA-SEATS (X-13A-S) Fotran program developed at the US Census Bureau [33]. One of the main motivations behind this endeavor was to allow easy access to performing the most used X-13A-S computational routines with the simple click of a mouse or keyboard button, and being able to visualize the results instantaneously. This initial program spawned into what is now called

*uSimX13* and is featured as one of the four model-based time series analysis modules of the entire iMetrica software package. In this document, we reveal the inner workings of the uSimX13 module: what it does, how it is organized and built, how to use it for analyzing time series, and how to use it for simulation studies of classical and modern time series diagnostic tests.

We believe that this is the first software of its kind - being able to model and investigate the dynamics of nonstationary time series data using diagnostics along with an array of real-time visual tools, all through the simple control of a unique graphical user interface. With the primary goal of analyzing economic time series data using X-13A-S and producing models, adjustments, and forecasts instantaneously without the need of writing input files or any intricate knowledge of how the time series methodologies work, the target audience of the uSimX13 package is primarily anyone with the desire to learn and/or investigate the modeling and dynamics of nonstationary time series. It is also quite useful for practitioners who wish to quickly test different model fits and adjustments or obtain forecasts or seasonal/trend adjustments of their data without resorting to input/output files. To test one's model fits and adjustments, the environment provides not only a large array of classical and modern goodness-of-fit (gof) signal extraction diagnostic tests to assess different model fits of the data but also the associated graphics of the measures used in the computation of the gof diagnostics to get a visual aid into how they are functioning.

With the most common objective of time series analysis, seasonal adjustment, and signal extraction being to better understand the underlying evolution of the time series, this interactive software offers an engaging and innovative approach to understanding these important computational tools of time series analysis. Some of the principle computational tools include seasonal ARIMA (SARIMA) modeling, seasonal adjustment, regression component adjustment, and forecasting, all boasting a fast computational core for instantaneous results. As with all the other iMetrica modules, the layout of the uSimX13 module user interface presents a very simple intuitive design that allows users to see the results of their time series computations automatically. Once parameter estimations have changed or model adjustments are made, the resulting signal extractions, forecasts, diagnostics, spectral domain functions are changed and/or computed autmatically and plotted on one of the many plot canvas'.

What makes this time series modeling and simulation environment fast and robust is that most of the computational core is written in Fortran and C with many of the time series subroutines borrowed from the X-13A-S software package. In order to access many of the routines in the X-13A-S Fortran software and to speed up the computational core, function wrappers have been written for the important subroutines to handle data and parameter passing, all while avoiding much of the file writing and unnecessary routine calling. However, we would like to stress that the uSimX13 environment was not designed with the purpose of creating an alternative to the X-13A-S software, but rather as a high-end user-interactive tool for time series modeling which allows the user to actually visualize the effects of the modeling and simulation controls, as well as

perform many of the most used features of the X-13A-S environment. However, many features in the X-13A-S program are yet to be implemented in uSimX13, such as the X11 seasonal ajustment method (SEATS is used as the method for seasonal adjustement, and signal extraction in general. See Gomez and Maravall [14]).

The design of the software emphasizes an engaging interaction with the modeling and designing of the SARIMA model specification so that the user has the ability to make slight adjustments to model parameterizations, random seeds for simulations, number of observations, and other features while at the same being able to visualize the effects of the adjustments right away. This provides a unique interactive learning tool for newcomers to economic time series modeling.

Another feature of the uSimX13 module that is highly attractive to researchers in time series analysis is the ability to perform Monte Carlo studies for time series diagnostics. Due to the fast computational power of the module and easy data storing and plotting capabilities, Monte Carlo simulation studies are fairly simple to carry out in uSimX13. As an alternative to using X-13A-S for Monte Carlo studies where the program must store the precomputed simulation data from an independent program in separate files which are then inputed into a spec file for the X-13A-S program to read, the uSimX13 environment relies on the many data structures and storage capacities in the Java Virtual Machine [21] to render Monte Carlo simulation studies fully interactive with the iMetrica software environment, without resorting to output or reading input files.

In this paper, we discuss some aspects of the software from the X-13A-S modeling subroutines featured in the uSimX13 computational engine to spectral domain tools, gof diagnostics, and SARIMA and regressor modeling. Although many of the features of the uSimX13 GUI components are rather primitive and self-explanatory, in section 4, we give examples of the modeling and simulation capabilities helpful for both beginners of nonstationary time series modeling and practitioners looking to seasonally adjust data or assess model fit using gof diagnostics and spectrum tools.

## 2 uSimX13 Design

Software such at MATLAB, R, and the Ox programming language [22] provide rich libraries for statistical time series analysis using both objective-oriented high-end language and an array of computational libraries for specializing in matrix computations. However, one typically needs to write the programs in their native language and this can be a time wasting task for someone who isn't familiar with programming in general.

This software is fundamentally different however in that we have placed an emphasis on understanding time series modeling through a coupling of both visual tools and modeling diagnostics, both of which are accessible automatically through interaction with the GUI. This includes a visualization of outputs from many X-13A-S modeling and diagnostic procedures, with no input/output files to store nor a knowledge of computer programming; all data and computational structures

of interest is stored automatically and organized for automatic graphical output and diagnostic checking. In this section we give an overview of the structure and design of the uSimX13 software that helps achieve these fast results.

## 2.1 Overall Hierarchy

Since much of the computational effort in the environment is done in Fortran and C using shared libraries, the time series computations are as fast as in the original X-13A-S Fortran code, but now with the added benefit of seeing the modeling results change in real-time as parameters change without additional script writing for each change in parameter made.

The overall design to the uSimX13 software takes a hierarchal approach in that most of the computing is done in Fortran and C, with the data, parameter, and other controlled input and output being handled by routines written in the Java programming language, with heavy emphasis on the AWT/Swing libraries for handling the graphical user interface. We show in Figure 1 the computational structure of the software.

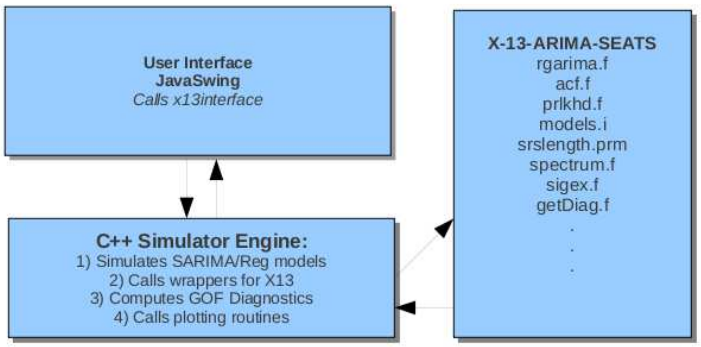


Figure 1: The overall design layout of the uSimX13 environment. All elements of the program are connected to a C program and input/output of time series data and parameters is handled through the Java GUI program.

The central interface is a GUI built using the Java Swing libraries which handles all the parameter adjustment controls for all the features of uSimX13. Java routines then handles the changes globally and passes the changes into wrappers written using the Java native interface library (jni) for calling all the C functions. The C functions then do the intermediary work, calling the necessary Fortran X-13A-S and C computational subroutines, and passing the computed data back into the Java interface. With the object-oriented Java interacting and passing parameters into wrappers for the X-13A-S calls, this hierarchical structure to the software allows for additional computa-

tional time series methods written in either C or Fortran to be added with relative ease. Since we expect many more features and improvements to be added in the near future to the uSimX13 software, careful consideration has been taken into account for the implementation and handling the communication between the uSimX13 GUI and the actual computational core.

## 2.2 Breakdown of the Computational Components

The breakdown of the computational components for uSimX13 has been done as follows. One of the main features of the environment is the ability to simulate time series data and employ all of the numerical routines on the simulated data. In order to minimize the inter-language communication effort between Java, C, and Fortran, the data simulation features have been written in the Java environment for direct access to the GUI. The simulated data is stored using Java data structure objects and passed into the C environment when needed. Spectral domain time series functions for the modeled data and the signal extraction and gof diagnostic tools (see section 4.2) are computed and sampled for plotting in a C library called `libSARIMAmodel.a`.

The ability to upload time series data from files is also possible. Using a file handling procedure built in the Java Swing library, one or several time series data files can be read and stored in the uSimX13 system. Details on formatting issues with the data files for uploading ones data is given in section 3.1.

Once data is stored in the uSimX13 system, all modeling components of the software are accessible and can be applied, visualized, adapted, and changed to fit the needs of the user. The three major components of the uSimX13 modeling environment are as follows. We will give more detail on each component later in the paper.

- *X-13A-S modeling environment*: The principal component of uSimX13 that features SARIMA model parameter estimation (MLE), signal extraction, seasonal adjustment, regression component extraction, forecasting, computation of both time and frequency domain gof diagnostics, AIC, BIC, and other model comparison indicators, regressor identification, and more.

- *SARIMA model simulation environment*: The controls for simulating time series data features the selection of SARIMA model orders for both nonseasonal and seasonal components. The number of observations, random seed for generating the random innovation process, the innovation variance of data generating process,

- *Pseudo-model estimation*: The ability to compute the pseudo-true models based on data from a fixed SARIMA model. This interface explores alternative models to time series data that is generated from a SARIMA model where the new parameters that are computed satisfy a minimization of a certain function called the Kullback-Leibler distance function.

These parameters are labeled pseudo-true and both the goodness-of-fit diagnostics and model comparison statistics using X-13A-S are computed using these pseudo-true values.

In conjunction with the various computational time series tools comes the many visualization tools to assist in analyzing the time series data. Both time and frequency domains are featured in the software for plotting

- original raw data
- seasonal component
- trend component
- cyclical component
- irregular component
- seasonally adjusted

in the time domain and

- periodograms
- spectral densities
- signal extraction weighting functions

in the frequency domain.

We mention again that all plotting features are dynamic in that once model choice/parameter choices are adjusted, new plots reflecting the modeling changes are automatically updated. We will summarize in more detail each of these modeling components throughout the remainder of this paper.

# 3 Outline of uSimX13 Module

In this section we give an overview of uSimX13 module interface and give a description of each of the features and controls. Figure 2 gives an overview the interface in the most recent release of the iMetrica software. Note that additional options and controls may have been made since the release of this documentation. Below we describe each of the panels in further detail.

1. *The Global menu for the uSimX13 module*: Handles the ability to import, export, and save different time series and signal extraction results.


Figure 2: The overall layout of the uSimX13 main control interface.

2. *Time series plots*: Control which time series are plotted on the plot canvas. By default the original time series is plotted. One can also plot (upon availability) the seasonally adjusted series, forecasts for up to 24 steps-ahead, the trend, the seasonal component, and the regression adjusted series if a regression option is used. To plot any of these series, simply click in the check box to activate the plot. Note that the seasonal component will be plotted such that the first point in the series represents the first observation in the original series.

3. *MLE and model comparison diagnostics panels*: When the MLE panel is chosen, all diagnostics and MLE values are computed and displayed relative to the model dimensions chosen using the combo boxes in the "model" box.

4. *Pseudo estimation panel*: (shown in Figure 3) the signal extraction estimations and model parameters are produced using the pseudo-true values, where the true (simulated or data generating process) model can be selected using the combox boxes in the *Model* panel.

5. *The Model* panel: Model selection panel for choosing dimensions of the SARIMA model. Note that the largest values currently available in the uSimX13 module for the model are $(2,1,2)_1(1,1,1)_{12}$. For models with higher dimensions, one must use either the *BayesCronos* [4] or *StateSpace* [5] modules which allow higher dimensions for the models. The regression

components can also be selected in this panel. To check for and compute a regression component, simply click the respective check box, then click the *regression adjusted* plot check box (in panel 2) to see the resulting adjusted data.

6. *Diagnostics panel*: Displays the signal extraction gof diagnostics (see 4.2), and four different information criteria (see 4.1). Changes in the model or regression components will reflect the new diagnostics instantly making quick model comparisons between different models a trivial task.

7. *The Ljung-Box statistics panel*: Features four different lags and prints the Ljung-Box statistics after any model estimation procedure (see 4.1 for more information on the Ljung-Box statistics).

8. *The MLE panel box*: The values of the model parameters are displayed when a model changes or a regression component is computed. When any adjustments are made to model dimensions or regression components added, the new MLE values are plotted instantaneously.

9. *The spectral plot panel*: Controls which spectral plots are to be plotted on the spectral canvas. To activate these plots, the spectral canvas must be chosen (choose in 11). The choices are (in clockwise order), periodogram $I_n(\omega)$, spectral density $F_{W,\psi}$, spectral differencing operator $D_n$, model spectral density times signal/model ratio $F_{W,\psi}g_\psi$, periodogram times signal/model ratio $I_n g_\psi$, signal/model ratio $g_\psi$.

10. *The signal/noise decomposition selector*: To select a certain signal, simply click on the desired check box. The choice of the signal only has effect on the spectral plot of the $g_\psi$ function which is the signal/model ratio in the frequency domain. Once selected, the new decomposition is reflected in the spectral canvas. It has no effect in the modeling or diagnostics procedures, only displays the different spectral plots for learning and analyzing purposes.

11. *Time/Frequency tabbed panels*: Choose between the time domain and the spectral/frequency domain. If the spectral domain is chosen, access to the different model decompositions and the spectral plots are available. If the time domain is chosen, the six different time series (original, seasonal, trend, forecasts, seasonal adjusted, and regression adjusted series) are available for plotting.

In the Pseudo-Estimation panel, additional diagnostics are featured that take into account two different models being compared. Figure 3 shows the panel and the features and diagnostics. As mentioned before, in the *Model* panel, two models can now be chosen offering the ability to compare models by assuming that the data is generated by a certain data generating process (DGP). The *Model* panel features two models for which one can choose the model dimensions 1)

The simulated (or DGP) model, and 2) the estimated model. For example, for any given time series data, if one assumes that the data is generated by an Airline model, comparing other models to the Airline model will produce diagnostics with the assumption that the data is generated from this "true" model. The estimated parameters for the *Estimated model* are called the *pseudo-true* model estimates and can be used for computing signal extraction efficacies and to be used in model comparison. For more information on pseudo-true values, signal extraction efficacies, and their applications, see section 4.2.2 for more details.

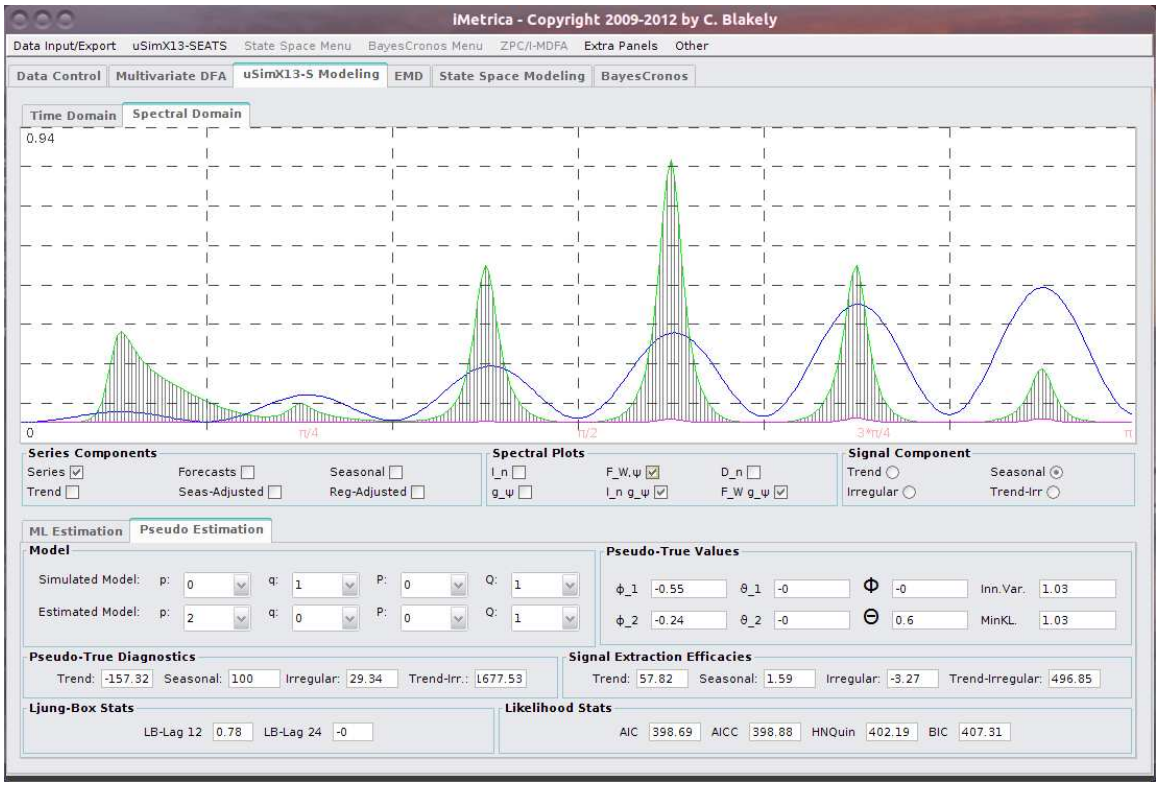


Figure 3: Pseudo-Estimation modeling panel in uSimX13. Offers the ability to compare models by assuming that the data is generated by a certain data generating process (DGP). The *Model* panel now features two different models to choose: 1) The simulated (or DGP) model, and 2) the estimated model.

## 3.1 Data Import/Export

There are various ways of inputing data into the uSimX13 module. The first option is to begin with the Data Control manager module. This module is the main data control hub for the iMetrica software and serves to collect time series data from a multitude of sources and then gives the user the ability to perform prior adjustments to the data such as scaling and log transformations. One also has access to the multitude of simulation of time series options. For further information on the

Data Control module and how to use it, please refer to [3]. Once data has been entered in the Data Control module, one can easily export it to the uSimX13 module by clicking on the Data Control main menu, and then clicking "Export to uSimX13". Once the uSimX13 module panel has been clicked, one will see the exported data already plotted in the plot canvas. Furthermore, the data is automatically modeled with the given settings in the uSimX13 environment, thus access to the trend, seasonal component, forecasts, are automatically available.

Of course, one has the option to import time series data directly into the uSimX13 module bypassing the Data Control center. The uSimX13 module menu features two options for doing this: by uploading a single time series data file, or by uploading a X-13A-S datameta file. To upload a single data file, simply click *Open File* in the uSimX13 module. From here, a file open dialog appears (see Figure 5). Data files are typically marked with an extension `.dat`, however, any file type can be used as long as it adheres to the X-13ARIMA-SEATS data file format. This file format is straightforwardly given by either one or two columns of time series data observations. If two columns are given, the first column is assumed to be the date of the given observation in the second column. This format is called the "datevalue" format in X-13A-S and more information can be found on this format in Hood and Monsell [19]. To filter out any files that are not data files, simply click *Files of Type* and select *Data Files*. Once the .dat file has been selected, the data files are automatically loaded into the module environment.

A second method of importing data into the module uSimX13 is via the datameta file. A datameta file is a text file that lists a sequence of time series data files. The datameta file typically has the file extension `.dta` and once loaded, gives easy access to a large number of series to be modeled in the uSimX13 environment. To load such a file, go to the uSimX13 menu (see Figure 6) and click *Open MetaFile*. This opens a file dialog box from which one can access the directory storing the .dta file. To filter out any files that are not data or datameta files, simply click *Files of Type* and select *Data Files*. Once the .dta file has been selected, the data files are automatically loaded into the module environment. To toggle which series to model, simply go to the *Observation Panel* in the *Additional Panels* menu and if successfully uploaded, the data files should be featured in the *Loaded Files* scroll bar (see 4 in Figure 4. To change files, simply click the left or right button of the scroll bar to access the different files uploaded. The file name will then appear in the text box.

Once the data has been modeled, saving the different components to output files is an easy task. For example, if seasonally adjusted data is to be exported to an output data file, simply go the uSimX13 menu and place the mouse over the *Save Series* submenu (see Figure 6). This will show a list of the different modeled components that one can save to an output file. To save the series, simply select the desired series from the list (if the series component is currently plotted on the canvas, it will become highlighted), and then click the left-mouse button. This will save the data in column format in the current directory where the iMetrica java class is found. The file

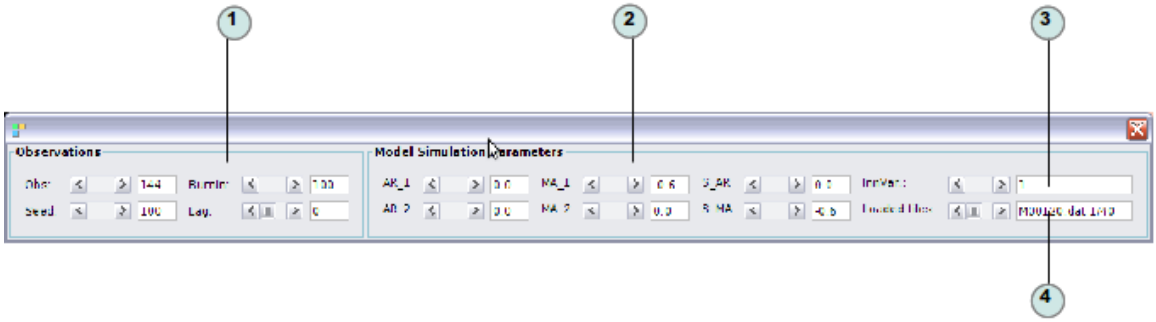


Figure 4: The additional panel featuring properties of the uSimX13 simulation data mode. The scroll bar 4 contains the time series data files currently loaded into uSimX13. To change files, simply click the left or right button of the scroll bar to access the different files uploaded. The file name will then appear in the text box.

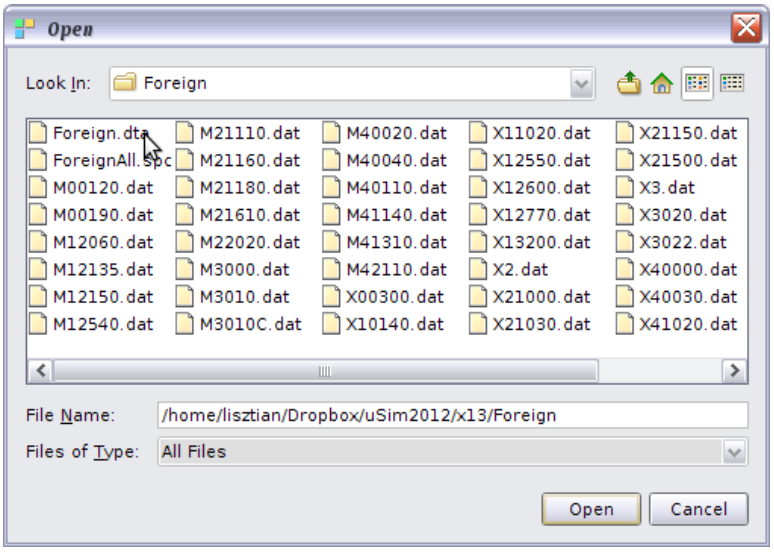


Figure 5: File Dialog box for opening .dat or .dta datameta file for opening a directory of time series files. The File Dialog can filter out all files without these extensions for an easier file search.

name will be `iMetrica-*.dat` where the asterix is the series type. For example, in the seasonally adjusted data case, this will be `seasAdj`. In the case of forecasted data, the `forecasted.dat` and includes the original time series with the 24 additional forecasted values appended to the end, thus it will be a file with $N + 24$ observations. Note however that the original dates of the time series, if supplied, are no longer retained in the file. Efforts might be made in future versions of iMetrica to include the original dates.

We also note that when another component series is saved, it replaces the older file with the same name. So if multiple seasonally adjusted data files are to be saved, each one must be renamed after each save before another one is stored.

One can also save different information and diagnostics produced by uSimX13 by again clicking on the uSimX13 menu and accessing the different *Save* menus. See Figure 7 for an example on saving the signal extraction component polynomials that are computed by SEATS.

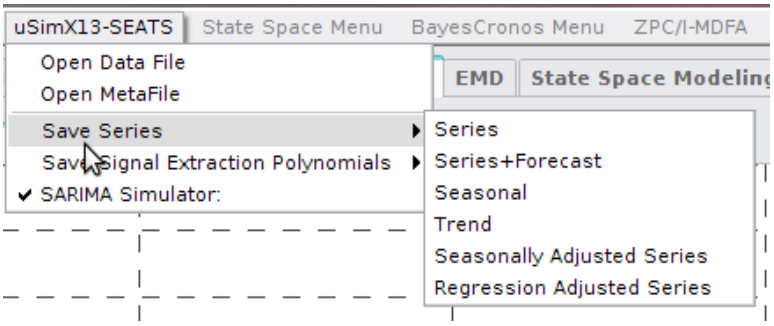


Figure 6: Menu for saving specific component series to an output file.

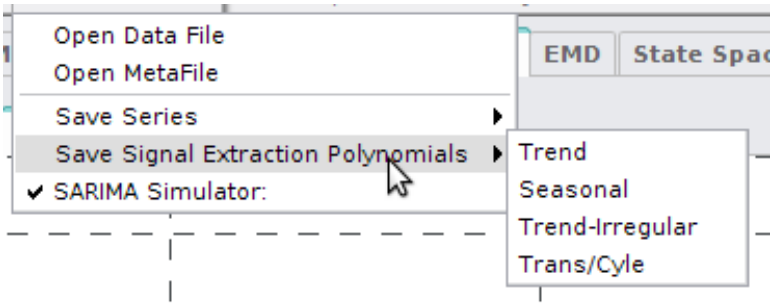


Figure 7: Menu for saving the canonical signal extraction polynomial coefficients produced by SEATS.

Before concluding this section on data import/export, we note a few tips when loading data into the uSimX13 environment. Firstly, when new data is imported into the environment, the data is automatically modeled using the given settings in the modeling environment. To make the adjustments between loading any data file smoother, we recommend that the default settings in the modeling environment be used. This implies using an Airline model (setting $p = 0, q = 1, P = 0, Q = 1$) and no regression effects. Secondly, be sure that the data being uploaded into the module is indeed data that would be appropriate for the X-13ARIMA-SEATS. Data that is stationary and non-seasonal could create internal problems and eventually slow down the module performance and the overall performance of the iMetrica software. For data that is nonseasonal, stationary, heteroskedastic, or anything else not typically modeled in X-13ARIMA-SEATS, the user is recommended to use any one of the other modules, depending on the goals and priorities of the user.

### 3.2 Simulation Environment

The uSimX13 module also provides an easy to use simulation environment for simulating time series from a model of the form $(p, 1, q)_1(P, 1, Q)_{12}$ where $p, q \leq 2$ and $P, Q \leq 1$. The simulator in uSimX13 is used primarily in the context of learning more about the dynamics of basic SARIMA models, pseudo-true model estimation, spectral density functions, and diagnostics.

By default, when the iMetrica software is started and the uSimX13 module is selected, the simulation environment is activated. To change the model dimensions of the data to be simulated, simply use the combo boxes in the *Model* panel shown in 2 of Figure 2 to choose set a value for $p, q, P, Q$, the nonseasonal and seasonal ARMA dimensions. Once the dimensions are set, the *Observation Panel* is to be opened to select the parameters used for simulation. To open the observation panel, go to to *Extra Panels* menu at the top of the iMetrica frame, and select *Observation Panel*. This opens an additional dialog frame from which one can choose the different parameters of the simulated data. Once a value has been adjusted, the new data is automatically plotted and modeled. For example, to change the MA parameter, simply click on the scroll bar next to $\theta_1$. The new value is reflected in the new data being plotted. To be sure, adjust the value and one should see that the MLE values change as well. One can also scroll through different random seeds that change the innovation sequence used in simulation, the number of observations, the burn-in period of the simulation, and the lag structure for the diagnostics (typically lag 0,1,or 12 is used). Figure 4 shows this *Observation Panel* in greater detail.

## 4 uSimX13 Methods

The X-12-ARIMA seasonal adjustment program is an enhanced version of the X-11 Variant of the Census Method II seasonal adjustment program. The X-13A-S program is an enhancement of the X-12-ARIMA program to include the signal extraction for ARIMA time series program SEATS designed by Maravall and Gomez at the Bank of Spain (see [14] for more information on the SEATS program). One of the main advantages of this fusion effort was to provide an even more versatile approach to modeling seasonal (or nonseasonal) time series using both an array of deterministic components to capture outliers, calender effects, and other regression effects and a variety of simple stochastic models to capture other structural components of the series, such as seasonality, trends, and cycles. uSimX13 provides access to many of the important methods in X-13A-S and in this section we summarize some of the features. Of course, for more information pertaining to each individual method, we recommend the literature mentioned at the conclusion of each subsection.

Much of the SEATS functionality in the uSimX13 module comes from the unobserved component canonical decomposition capabilities. For example, the uSimX13 program produces seasonal adjustments that are based on the output of the seasonal component that is produced in the SEATS

signal extraction estimation routines. The functionality borrowed from X-12-ARIMA comes from the *regARIMA* routine that simultaneously estimates an ARIMA model (whether with fixed dimensions or using *automdl* and computes any regression components (if any).

As mentioned in the introduction, a substantial effort in creating the uSimX13 environment was aimed at providing not only robust and state-of-the-art time series computational tools, but also to provide them in a computationally fast manner. Utilizing the construction of wrappers in Fortran and C, the X-13A-S subroutines used in uSimX13 have been streamlined by avoiding unnecessary calls to file writers and readers, while trying to reduce the amount of parameters in the X-13A-S function calls. This ultimately speeds up the necessary function calls and nested function calls. For example, we show in Figure 4 the network of function and subroutine dependencies within X-13A-S. The arrows show the direction in which a function calls another function. In this case, the `rgarima.f` function of X-13A-S computes the regression and ARIMA model parameters for a given data set and user inputs.

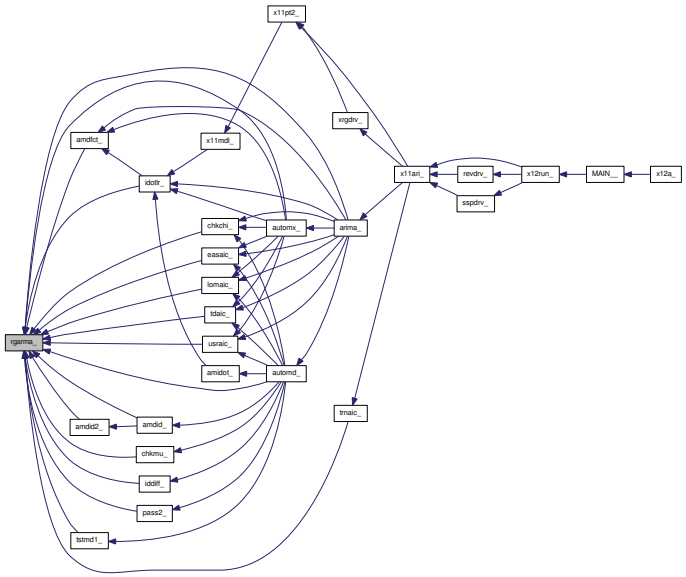


Figure 8: The network of function and subroutine dependencies within X-13A-S when calling the rgarima.f function used for model estimation. The arrows show the direction in which a function calls another function.

However, as one can clearly see, in order to get to this function, a link to many different function calls must be executed prior to any model estimation rendering many computational inefficiencies in X-12. In uSimX13, we attempt to bypass much of the function linkage to `rgarima.f` and simply access the numerical algorithms of the estimate routines more directly through the use of function and subroutine wrappers.

We first look at the general model and estimation component borrowed from the X-12-ARIMA and SEATS libraries.

### 4.1 Features from X-13A-S: General Model and Estimation

The uSimX13 environment centers around modeling seasonal (or nonseasonal) nonstationary time series using seasonal ARIMA (or SARIMA) models. As discussed by Box and Jenkins [7], ARIMA models are frequently used for seasonal time series and generally, considerably easy to estimate using both standard maximum likelihood techniques being used by X-13A-S or Bayesian techniques (we only discuss the former in this paper). A general multiplicative seasonal ARIMA model for a time series of monthly data $Y_t$ can be written, after suitable Box-Cox transformations, as the following:

$$\phi(B)\Phi(B^{12})(1-B)^d(1-B^{12})^D Y_t = \theta(B)\Theta(B^{12})\epsilon_t^Y, \quad t = 1, 2, \ldots, N \tag{1}$$

where $\phi(z), \Phi(z^12), \theta(z)$, and $\Theta(z^12)$ are polynomials of degree $p, P, q, Q$ respectively, with roots outside the unit circle of the complex plane, ensuring an invertible representation for the differenced data. The $\epsilon_t^Y$ is a white noise innovation sequence (we emphasize the superscript $Y$ since we will have other innovation sequences for different processes). Here, $B$ is the standard backshift operator, namely $BY_t = Y_{t-1}$. We note that we allow for $d$ nonseasonal differences and $D$ seasonal differences, however the uSimx-13 signal extraction program routines currently work only for $d, D \leq 1$, but not both zero). We will typically denote this model as a SARIMA $(p, d, q)(P, D, Q)_{12}$ model. We assume that $Y_t$ can be rendered stationary by the differencing operator $\delta(B) \equiv (1-B)^d(1-B^{12})^D$, thus giving stationary data $W_t = \delta(B)Y_t$. We will let $W = \{W_1, W_2, \ldots, W_n\}'$ denote the available differenced data throughout the remainder of the paper.

One can choose an ARIMA model for the given data using the *Model* panel, or allow uSimX13 to automatically choose a model using the X-13A-S function *automdl*, a specification native to X-13A-S (see [33]). For automatic identification of the model, X-13A-S determines the orders for the SARIMA model (called ARIMA model identification) by following well-established procedures that rely on examination of AICs, the sample autocorrelation function (ACF) and sample partial autocorrelation function (PACF) of the time series $Y_t$ and its differenced data $W_t$. It is based on the procedure in the TRAMO-SEATS time series modeling program developed by Gomez and Maravall (see [14], [15], and [16]) but contains modifications to make use of the native X-13A-S model estimation procedure, transformation and outlier identification procedures, and model diagnostics.

Once the order of the model has been determined, uSimX13 calls the **estimate** procedure in the `rgarima.f` that estimates the model parameters by exact maximum likelihood, or by a variant known as conditional maximum likelihood (Box and Jenkins [7]), which is sometimes called conditional least squares. The default in uSimX13 is the exact maximum likelihood approach for both the AR and MA parameters. The resulting log-likelihood for an ARIMA model is reduced to a sum of squares function minimized in a nonlinear least squares routine provided by MINPACK discussed by More, Garbow, and Hillstrom [13].

In the case that both ARIMA model and regression components have been specified in the *Model* panel (also known as a *regARIMA* model), the approach to maximize the likelihood uses an iterative generalized least squares (IGLS) algorithm of Otto, Bell, and Burman [30]. This algorithm involves two general steps that for any given values of the AR and MA parameters, the regression parameters that maximize the likelihood are obtained by a generalized least squares (GLS) regression. Then, for given regression parameters, the ARIMA model is then computed by maximum likelihood to the time series of regression

errors. IGLS iterates between these two general steps until convergence is achieved.

Once the regARIMA model has been computed, one can visualize the effects of the regression component by plotting both the original data and the regression adjusted data in the plotting canvas (see 2 in Figure 2).

#### 4.1.1 Information Criterion

The uSimX13 program provides the following model selection information criteria each time model estimation of time series data is performed: AIC (Akaike [1], see also Findley [10]), AICC (Hurvich and Tsai [20]), a criterion due to Hannan and Quinn [17], and BIC (Schwarz [31]). These numbers draw useful information during the course of model comparison and are used in the model estimation stage for user inputed data. Suppose the number of estimated parameters in the model, including the variance of the innovation process $\epsilon_t^Y$ is $n_r$ and that $N$ is the length of the data after applying the differencing operators. Denote $L_N$ the estimated maximum value of the exact log likelihood function of the model for the untransformed data, then the formulas for these criteria are:

$$\begin{aligned} AIC_N &= -2L_N + 2n_r, \quad AICC_N = -2L_N + 2n_r(1 - \frac{n_r+1}{N})^{-1} \\ HQ_N &= -2L_N + 2n_r \log\log N, \quad BIC_N = -2L_N + 2n_r \log N \end{aligned} \tag{2}$$

Between any two models, Akaike's Minimum AIC criterion states that the model with the smaller AIC is preferred. Similarly, for each of the other model selection criteria above, the model with the smaller value is preferred. The reader is referred to Findley [10] for uses and limitations of the information criterion.

#### 4.1.2 Ljung-Box Test

The Ljung-Box test [23] is a classical gof diagnostic commonly used in modeling ARIMA models. It can be defined by the following hypothesis test:

- $H_0$: The data is random, any observed correlations in the data result from randomness of the sampling process of the data.

- $H_a$: The data is not random.

The test statistic is given by

$$Q = n\,(n+2)\sum_{k=1}^{h}\frac{\hat{\rho}_k^2}{n-k} \tag{3}$$

where n is the sample size, $\hat{\rho}_k$ is the sample autocorrelation at lag $k$ of the residual between the model and data, and $h$ is the number of lags being tested. For significance level $\alpha$, the critical region for rejection of the hypothesis of randomness is $Q > \chi^2_{1-\alpha,h}$, with the right hand side of the inequality the $\alpha$-quantile of the chi-square distribution with $h$ degrees of freedom.

So if the model is correct, the Ljung-Box Q-statistics are asymptotically distributed as $\chi^2$ with degrees of freedom equal to the number of lags used in computing them less the number of AR and MA parameters estimated. The $p$-values of the Ljung-Box statistic approximate the probability of observing a Q-value at least this large when the model fitted is correct. When the degrees of freedom is positive, small values of $p$, customarily those below 0.05, indicate model inadequacy.

In the uSimX13 program, the Ljung-Box Q-statistic p-value for a given estimated model and data can be computed at any lag up to 24 using a scroll bar to adjust the $k$ value indicating the lag. In many cases, the Q-statistic can provide very good results at lags 12 and 24 for seasonal data. However, it is often documented in the literature that this standard gof test fails to capture model misspecification pertinent to the task of seasonal adjustment. In the next section, we discuss a new genre of gof signal extraction diagnostics available in uSimX13 that were recently shown to yield better results for model misspecification.

#### 4.1.3 Forecasts

For a given SARIMA model with parameters estimated by the `rgarima.f` subroutine of X-13A-S, forecasting the data up to 24 steps-ahead can be computed using the estimated model. The point forecasts are minimum mean squared error (MMSE) linear predictions of future $Y_t$ values based on the present and past $Y_t$ assuming that the specified ARIMA orders are correct, and that the parameter values used are equal to the true values. These are standard assumptions, though obviously unrealistic in practical applications.

#### 4.1.4 SARIMA Model Dynamics

An innovative feature of the uSimX13 program that takes advantage of the computational efficiency offered by the optimization of the X-13A-S subroutine calling described earlier is the ability to understand better the dynamics of SARIMA models and the estimation of their parameters using Maximum Likelihood Estimation procedures. In this subsection, we explore the mechanisms of this uSimX13 feature and discuss how it can be used to learn and gain insight into SARIMA modeling for nonstationary time series.

The main engine of uSimX13 is the ability to efficiently simulate data from SARIMA models and component models. Consider simulating data from a $(p,1,q)(P,1,Q)_{12}$ SARIMA model as follows. Let $p,q \leq 3$, $P,Q \leq 1$, and $\sigma^2 > 0$ and let $(Y_0, Y_1, \ldots, Y_k)$ be $k$ arbitrary initial values where $k = p+12P+q+12Q+13$. If $\phi = (\tilde{\phi}_1, \ldots, \tilde{\phi}_{p+12\cdot P+13})$ and $\theta = (\tilde{\theta}_1, \tilde{\theta}_2, \ldots, \tilde{\theta}_{q-12\cdot Q})$ are the coefficients of the polynomials $\tilde{\phi}(z) = \phi(z)\Phi(z)(1-z)(1-z^12)$ and $\tilde{\theta}(z) = \theta(z)\Theta(z)$ from a given SARIMA model, then we can easily simulate a series $Y_t$ for $t = 0, \ldots, N$ from the $(p,1,q)(P,1,Q)_12$ by first simulating the white noise process $\epsilon_t^Y \sim N(0,\sigma^2)$ for $t = 0, \ldots, q+12Q+N+N_B$. Here, the integer $N_B$ is a so-called *burn-in* length for the simulation process, with the burn-in period allowing the effects of the SARIMA parameters to fully be captured in the simulated data. With the sample white noise process computed, the value of the data $Y_t = \phi \cdot (Y_{t-1}, \ldots, Y_{t-p+12P+13}) + \theta \cdot (\epsilon_t^Y, \ldots, \epsilon_{t-q+12\cdot Q}^Y)$ for $p+12P+13 \leq t \leq q+12\cdot Q+N+N_B$, where $\cdot$ represents the dot product of the two vectors. Once all values of $Y_t$ have been computed, the final $N$ values of the data are taken to represent the simulated of the SARIMA process. The default value for the burn-in period is $N_B = 100$ and the default initial values for $(Y_0, \ldots, Y_k)$ are zero. One can toggle the value of $N_B$, however, we have found through size and power Monte Carlo studies of the simulated data estimated parameters that the initial values have little or no effect on the quality of the simulated data given a reasonable burn-in period, so we have left these values at zero in the uSimX13 program.

Parameter changes can be made to the SARIMA model by use of scrollbars indicating the parameter values. Once a parameter has been adjusted, the SARIMA model simulated data is instantly replotted in the time domain plot box. For a fixed $N$, $N_B$ and a fixed random seed for the random number generator, small changes in a parameter are reflected automatically in the simulated data and small adjustments can be seen in the trend, seasonality, or white noise variance depending on which parameter has been adjusted. Furthermore, the effects of adding additional AR or MA parameters to the data can be visualized creating a unique environment to learning about how the SARIMA model parameters influence the dynamic structure of the modeled time series. Other factors available for adjusting the model or changing the simulated data include the number of observations $N$, the burn-in period $N_B$, and the random seed for the innovation process. Figure 4.1.4 shows the control panel for the SARIMA model parameters that control the exact values for the simulated data.

When adjustments in the SARIMA model are toggled, the uSimX13 program has the ability to also compute the MLE estimation of the data, AIC and other information criterion statistics, the Ljung-Box Q-statistic, and the gof signal extraction diagnostics described in the next subsection. This provides a practical approach to understanding how diagnostics change when small changes in the data occur. Other data of interest capable of being computed and plotted on the time domain plot box as small adjustments are made to the model are a forecast with up to 24 steps-ahead and the seasonal and trend components from the signal extraction procedure. On the spectral domain, spectral density functions of the associated data model, signal, and noise components can

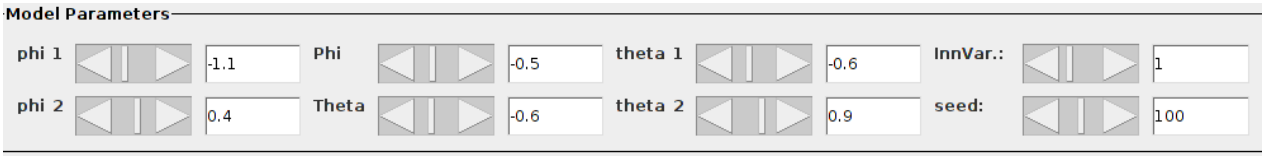


Figure 9: Once changes to one of the values has been made, the data is re-plotted keeping all other parameters, including the random number generator seed, fixed. This way, one is able to see how the dynamics of the underlying SARIMA model fluctuate according to small changes in the parameters.

be visualized as well.

### 4.1.5 Regression Components

A useful feature of the uSimX13 module is being able to easily test for and remove elements of the time series that cannot be modeled by a stochastic model component, such as outlier effects and calendar effects such as trading day and holiday effects. Calendar effects are quite often seen in monthly economic data and are produced by changing calendar effects such as days of week in a particular month or a moving holiday such as Easter or Thanksgiving. These features are modeled by using regression effects that are defined by a vector of coefficients $\beta \in \mathbb{R}^m$ that is computed in the estimation process, and a matrix $x_{i,t}$ of the regression variables defined by the $i$-th component at time $t < T$.

As mentioned before, models which feature both the SARIMA model and a regression component are called *regARIMA* models and their specification requires specification of both the regression variables (the $x_{i,t}$'s and a model for the error $z_t := y_t - \beta x_{i,t}$. The former is done by using one of the four available regression options in the *Model* panel (see 2 in Figure 2) and the SARIMA model is specified using the model dimension combo boxes also in the *Model* panel. Choosing which regression variables to include requires user knowledge relevant to the time series being modeled. Several regression variables that are frequently used in modeling seasonal economic time series are built in the uSimX13 module and include the following.

- Box-Cox transform: Although not a regression variable, it is however important in transforming data that grows exponentially as time increases, into a more "linear" process.

- Trading day regressor: Trading-day effects occur when a series is affected by the differing day-of-the-week compositions of the same calendar month in different years. Trading-day effects can be modeled with 7 variables that represent a specific number of days (no. of Mondays), . . . , (no. of Sundays) in a month $t$. . Bell and Hillmer [2] propose a better parameterization of the same effects that use 6 contrast variables defined as (no. of Mondays)-(no. of Sundays),

. . . , (no. of Saturdays) - (no. of Sundays), along with a seventh variable for length of month. See [33] Section 4.3 for more information on it's computation.

- Outlier effects: The outlier regressor is a quite versatile option which checks for and computes three different types of outliers in the data. Namely, Additive Outlier (AO), Level Shifts (LS), and Temporary Changes. AOs affect only one observation in the time series whereas LSs increase or decrease all observations from a certain time point onward by some constant amount. TCs allow for an abrupt increase or decrease in the level of the series that returns to its previous level exponentially rapidly.

- Easter day regressors: Holiday effects (in a monthly flow series) arise from holidays whose dates vary over time. They are significant enough if the activity measured by the series regularly increases or decreases around the date of the holiday and if this differentially affects two (or more) months depending on the date the holiday occurs each year. Easter effects are the most frequently found holiday effects in U.S. economic time series, since the date of Easter Sunday varies between March 22 and April 25. The Easter effect assumes that the level of activity changes on the $w$-th day before the holiday for a specified $w$, and remains at the new level until the day before the holiday. In uSimX13, one can compute the Easter regressor effect by clicking the Easter regressor box featured in the *Model* panel and then choose $w$ in the combo box.

More regression options, such as other holiday regressors, are continuously being added in uSimX13 and will be available in future distributions of the iMetrica software. This includes a date defined ramp regressor which model sudden or monotonic jumps in the level of the data.

A nice feature of the uSimX13 model is that not only are the regression components computed automatically when the respective regression component box is checked, but one can visually check for the significance of a specific outlier simply by toggling on/off the given effect using the check box. In the time series plots box, make sure that both the *original time series* and *regression adjusted* time series plots are both selected. When all regression components are checked off, the plot of the regression adjusted data will be the same as the original series. When a regression component is selected, on can immediately tell if it was significant by the plot color differences: if the regression component was significant, there will be a visual significant difference in the plots. One can then play around with different model dimensions and compare information criteria and signal extraction gof diagnostics to find the best model combination for the data at hand.

## 4.2 Additional Features: Signal Extraction and GOF Diagnostics

Model-based signal extraction comprises a large role in the uSimX13 software and is handled through subroutines of SEATS and a C library. Since it forms the basis for the goodness-of-fit

diagnostics that will be discussed later, we give a brief summary of the definitions and terminology typically used in signal extraction theory and practice and what takes place during the extraction process after SARIMA model estimation process of section 4.1 has taken place.

We begin by first giving a few definitions. Consider a stationary series $\{W_t\}$. The periodogram of $\{W_t\}$ is defined by

$$I(\lambda) = \frac{1}{n}\Big|\sum_{t=1}^{n} W_t e^{-it\lambda}\Big|^2 = \sum_{h=1-n}^{n-1} R(h)e^{-ih\lambda},\ \lambda \in [-\pi, \pi],$$

where $R(h)$ is the sample autocovariance function and $i = \sqrt{-1}$. Furthermore, we can define the spectral density $f_W$ as

$$f_W(\lambda) = \sum_{h=-\infty}^{\infty} \gamma_{f_W}(h)e^{-ih\lambda}$$

so long as the autocovariance function (ACF) $\gamma_{f_W}(h)$ is square summable. For any bounded positive symmetric function $g(\lambda)$, we define its inverse Fourier Transform by

$$\gamma_g(h) = \frac{1}{2\pi}\int_{-\pi}^{\pi} g(\lambda)e^{ih\lambda}d\lambda,\ h = 0, \pm 1, \pm 2, \ldots$$

With these definitions, we can now layout the signal extraction methodology used in uSimX13. The first assumption is to assume that the nonstationary time series $Y_t$ can be decomposed into a possibly nonstationary signal component $G_t$, the signal of interest, and a possibly nonstationary noise component $N_t$, all other components grouped together, such that $Y_t = G_t + N_t$. With this decomposition, our second assumption is that the differencing polynomial $\delta(z) \equiv (1-B)^d(1-B^{12})^D$, can be factored into polynomials $\delta^G(z)$ and $\delta^N(z)$ which share no common zeros and such that $U_t = \delta^S(B)G_t$ and $V_t = \delta^N(B)N_t$ are now mean zero stationary signal and noise components that are uncorrelated with one another. It follows now that since $Y_t = G_t + N_t$, we have that $W_t = U_t + V_t$ is stationary.

The next step is to first identify a SARIMA model 1 for the data $Y_t$. (In the next section we give an outline of how to do this in an efficient manner in uSimX13 based on Likelihood information criterion such as the AIC). Once model identification has been accomplished, the Maximum Likelihood Estimation procedure is then used to compute the seasonal and nonseasonal AR and MA parameters. Going back to the definition of the spectral density for $W_t$, clearly there is a one-to-one mapping from the space of parameters of the SARIMA model for $Y_t$ to the spectral density function $f_W$. Thus we can suppose that $f_W$ belongs to a family of spectral densities that is parametrized by a vector $\psi = (\psi_1, \psi_2, \cdots, \psi_r)'$, where $\psi_1, \ldots, \psi_{r-1}$ are the AR and MA parameters and $\psi_r$ is the innovation variance $\sigma^2$ for the white noise process $\epsilon_t^Y$.

Once the SARIMA 1 has been identified and estimated, the next step in signal extraction begins by specifying models for each of the signal and noise components. We will follow the canonical decomposition approach of Hillmer and Tiao [18], since this approach is the one used in SEATS.

The assumed component models for the signal component $G_t$ can be given by a seasonal, trend, or irregular component or a combination of any two. These components are defined as

$$\begin{aligned} \Phi(B^{12})U(B)^D S_t &= \theta^S(B)\epsilon_t^S \\ \phi(B)(1-B)^{d+D}T_t &= \theta^T(B)\epsilon_t^T \\ \phi^I(B)I_t &= \theta^I(B)\epsilon_t^I, \end{aligned} \tag{4}$$

where the various MA polynomials $\theta^{\cdot}(z)$ are chosen to guarantee invertible representation and the $\epsilon_t^{\cdot}$ innovation sequences are independent white noise. The polynomial $U(z) = 1 + z + z^2 + \cdots + z^{11}$ achieves seasonal differencing and satisfies $1 - z^{12} = U(z)(1 - z)$. The irregular component $I_t$ is a quite versatile component and in SEATS parlance is labeled a transitory component. In most cases, the irregular component will simply be assumed a white noise process, thus $\phi^I(z) = \theta^I(z) \equiv 1$. However, depending on the location of the roots for the AR and MA components of the fitted SARIMA process, SEATS occasionally will attempt to model a cyclical-trend component, which in effect defines $\theta^I(z)$ to be a polynomial of degree 3.

The estimation of the parameters for the MA polynomials $\theta^{\cdot}(z)$ and the innovation variance $\epsilon_{\cdot}^2$ is accomplished in a SEATS subroutine called *spectrum.f* that computes a partial fraction decomposition and outputs the polynomial coefficients of the MA polynomials. The reader is referred to the original paper by Hillmer and Tiao [18] for a detailed description of the approach. Here we outline an example to highlight the total procedure from estimation to signal-noise decomposition.

### 4.2.1 Example

Consider a SARIMA $(0,1,1)_1(0,1,1)_{12}$ model that, after seasonal and nonseasonal differencing with $\delta(B)$, can be decomposed into a signal component $G_t$ given by the sum of trend and irregular. Since $\delta(B) = (1-B)(1-B)^{12} = U(B)(1-B)^2$ and since $I_t$ is stationary, we associate the differencing $(1-B)^2$ with the signal $T_t + I_t$ to reduce it to stationary, which then gives $U_t = Z_t + (1-B)^2 I_t$. Here $Z_t = (1-B)^2 T_t$ is stationary, and actually follows an MA(2) model. Denoting this MA(2) polynomial by $\xi(z) = 1 + \rho_1 z + \rho_2 z^2$, it then follows that we can then write the spectral density of the process $U_t$ as

$$f_U(\lambda) = |\xi(z)|^2 \rho_3 + |1 - z|^4 \rho_4,$$

where $\rho_3$ and $\rho_4$ are the innovation variances of the trend and irregular processes.

The noise component will then be given by a seasonal component, and therefore the noise differencing operator is $\delta^N(z) = 1 + z + z^2 + \cdots + z^{11}$ resulting in the MA(11) process $V_t = \delta^N(B)G_t$ for the noise component. Now since $W_t = \delta^N(B)U_t + \delta^N(B)V_t$, we have

$$f_W(\lambda) = |\delta^N(e^{-i\lambda})|^2 f_U(\lambda) + |\delta^S(e^{-i\lambda})|^2 f_V(\lambda). \tag{5}$$

With the spectral density of $W_t$, it is clear that the parameters of $f_U$ and $f_V$ are direct functions of $\psi$, and are referred to as $\rho = \rho(\psi)$ and $\tau = \tau(\psi)$ respectively. Note that $\rho$ and $\tau$ typically have

different dimensions from $\psi$. We will write the spectral densities using the relationship between the parameters $\psi$ and the spectral densities functions of $f_U$ and $f_V$ as $f_{U,\psi}$ and $f_{V,\psi}$, while omitting reference to the unknown mappings $\rho$ and $\tau$.

With the decomposition of $W_t$ into the proposed signal and noise components, we can now employ the X-13A-S gof signal extraction diagnostics to assess the fit of the SARIMA model to the data $Y_t$. The X-13A-S gof signal extraction diagnostics takes the following form:

$$Q(f, \psi) = \frac{1}{2\pi} \int_{-\pi}^{\pi} g_\psi(\lambda) f(\lambda)\, d\lambda$$

where $f$ is an integrable (possibly random) function on $[-\pi, \pi]$, and $g$ is a particular weighting function $g$ that depends on the parameter $\psi$ of the underlying SARIMA model. Signal extraction diagnostics (see Maravall [24], Findley, McElroy, and Wills [10], and McElroy [27]) typically introduce a weighting function that is heuristically the squared ratio of signal spectrum $f_S$ to data spectrum $f_Y$, which is designed to penalize deviations from the correct model more highly if they impact the frequencies associated with the signal. Therefore, using the spectral densities for the signal and the data, along with the noise differencing operator, the weighting function $g_\psi$ is defined by

$$g_\psi(\lambda) = \frac{f_{U,\psi}^2(\lambda) |\delta^N(e^{-i\lambda})|^2}{f_{W,\psi}^2(\lambda)}. \tag{6}$$

With the definition of the weighting function $g$ and the form $Q$, we can then consider two different quantities, $Q(I, \widehat{\psi})$ and $Q(f_{W,\widehat{\psi}}, \widehat{\psi})$. The first quantity of the two defines the signal extraction diagnostic, and clearly depends on the definition of the model $f_U$ (choice of signal) and $\delta^N(B)$ (choice of noise differencing operator). When we assume that the stated model is correctly specified or contains the correct model as a special case, the second quantity is the centering for this diagnostic and is computable since $\widehat{\psi}$ is the MLE of the SARIMA model. With these two quantities, the normalization of the difference $Q(I, \widehat{\psi}) - Q(f_{W,\widehat{\psi}}, \widehat{\psi})$, is then given by the quantity

$$V(\psi) = \frac{1}{\pi} \int_{-\pi}^{\pi} r_\psi^2(\lambda) d\lambda - 2b'(\psi) M_f^{-1}(\psi) b(\psi) \tag{7}$$

where

$$\begin{aligned}
r_\psi(\lambda) &= g_\psi(\lambda) f_{W,\psi}(\lambda) \\
b(\psi) &= \frac{1}{2\pi} \int_{-\pi}^{\pi} g_\psi(\lambda) h_\psi(\lambda) d\lambda \\
h_\psi(\lambda) &= \nabla_\psi \log f_{W,\psi}(\lambda) = \frac{\nabla_\psi f_{W,\psi}(\lambda)}{f_{W,\psi}(\lambda)} \\
M_f(\psi) &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \nabla_\psi f_{W,\psi}(\lambda) \nabla'_\psi f_{W,\psi}(\lambda) f_{W,\psi}^{-2}(\lambda) d\lambda
\end{aligned}$$

In the expression for $V(\psi)$, the second term encompasses the contribution of parameter uncertainty, since it involves derivatives of the spectral density $f_W$.

Since we wish to test model mis-specification from an imposed SARIMA model under the signal and noise components, we can use natural null hypothesis $H_0$ is that the model is correctly specified whereas the alternative $H_a$ posits that the model is incorrectly specified. The normalized test statistic is then defined via

$$T_n = \sqrt{n}\,\frac{Q(I,\widehat{\psi}) - Q(f_{W,\widehat{\psi}},\widehat{\psi})}{\sqrt{V_0(\widehat{\psi})}}, \tag{8}$$

which will be asymptotically standard normal under $H_0$. Thus when assessing model gof, it is important to remember that significant deviations of $T_n$ from zero indicate rejection of the null hypothesis, namely that the model specification is incorrect. This indicates that our data model is wrong, and hence the component models are also wrongly specified since they are derived via fixed algorithms from the data model. However, if the absolute value of the test statistic $T_n$ is close to zero, namely less than 1.96, then the data model is correctly specified (using a double sided test with $\alpha = .05$). The Blakely and McElroy [6] paper gives a large suite of empirical size and power studies confirming these results.

To gain a better understanding of the signal extraction computation, a feature of uSimX13 allowing one to visualize the spectral density functions of the signal extraction process has been integrated into the spectral domain functionality. For any given differenced time series data and corresponding SARIMA model fit, the functions $g_{\widehat{\psi}}(\lambda)I(\lambda)$ and $g_{\widehat{\psi}}(\lambda)f_{W,\widehat{\psi}}(\lambda)$ are plotted on the interval $[0,\pi]$ sampled at 500 points. The difference $g_{\widehat{\psi}}(\lambda)(I(\lambda) - f_{W,\widehat{\psi}}(\lambda))$ is also plotted using a dotted line. With a correctly specified model for the data $W$, the integral over $[-\pi,\pi]$ of the difference should be relatively close to zero, with the spectral peaks of both spectral densities correlating with each other.

To summarize this section, a general outline for computing the gof diagnostic test is given as follows:

1. Begin with a proposed SARIMA model for $Y_t$. Derive the expression for the spectral density $f_{W,\psi}$ of the differenced process $W_t$. This can be plotted in the spectral panel.

2. Establish or specify the stationary components $U_t$ and $V_t$ of the signal and noise, and derive formulas $f_{U,\psi}$ and $\delta^N(z)$. These can be selected in the signal spectral density panel (see Figure 2).

3. Estimate the model parameters $\psi \in \Psi$ to obtain the estimate $\widehat{\psi}$, and determine the corresponding component parameters $\widehat{\tau} = \tau(\widehat{\psi})$. This procedure is performed automatically when a signal has been selected.

4. Compute $Q(I,\widehat{\psi}) - Q(f_{W,\widehat{\psi}},\widehat{\psi})$ and the variance estimate $V_0(\widehat{\psi})$ under the null hypothesis that the model is correctly specified. Both quantities can be plotted in the spectral signal/noise panel (see 10 Figure 2).

5. In the definition of $g_\psi$, we insert the function $\cos(k\lambda)$ for $k = 0, 1, 2, \ldots, 24$, which is referred to as the "lag $k$ diagnostic" for short. The lag can be toggled in the menu additional control panel found in the additional panel menu above the plotting canvas.

We end this subsection by making a few remarks concerning the computation of the gof diagnostic in uSimX13. The current version of the software computes and displays the diagnostics using the trend $T_t$, seasonal $S_t$, irregular $I_t$ and trend-irregular $T_t + I_t$ signal components while a transitory component $C_t$, if computed by SEATS, is associated with the irregular component. The lag $k$ diagnostic for any integer up to 24 for each of these signal decompositions can be computed as well. While all four diagnostics at any lag can yield different results, with some being outside threshold of 1.96 and some being close to zero, it was demonstrated through empirical studies in the paper by Blakely and McElroy [6] that the most consistent of the signal and lag choices were the seasonal and the trend-irregular at lag 0 and 12 for seasonal data. However, empirical results for real US Census Bureau economic data, as opposed to simulated data used in the size and power studies, demonstrated that seasonal, trend, and trend-irregular signal components yielded favorable results depending on the type of data being analyzed.

### 4.2.2 Pseudo-True Parameters and Model Fitting

In addition to the standard MLE process for estimating parameter values of a given SARIMA model, an additional toolkit has been integrated into uSimX13 based on recent research of signal extraction diagnostics and model misspecification. For given data and a postulated SARIMA model for the data, a *pseudo-true* parameter estimator for a given alternative SARIMA model to the postulated is available. Recall that the pseudo-true values for parameters are the parameter values that minimize a certain distance to another non-nested fixed model. The distance function used is the so-called Kullback-Leibler information divergence function [9].

In uSimX13, the key motivation in computing pseudo-true values arrives when one considers the meaning of a gof signal extraction diagnostic test when the test statistic reveals itself significant. For a given seasonal time series data set, a proposed SARIMA model, and a signal/noise component decomposition, then 1) what alternative models should be proposed based on the diagnostics, 2) can the model selection be based simply on the magnitude of the test statistic. Pseudo-true can perhaps be used to investigate these two questions related to the gof diagnostic.

To define the pseudo-true values, we first define the Kullback-Leibler information divergence (or discrepancy). Although not a true metric, the KL divergence measures the distance between the true model of the data, and a proposed "ansatz" approximate model. It is defined as

$$D(k, h) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \left( \log k(\lambda) + \frac{h(\lambda)}{k(\lambda)} \right) d\lambda,$$

(see Dahlhaus and Wefelmeyer [9]). With $\tilde{f}$ denoting the true spectral density of the data $W$, the

pseudo-true value $\tilde{\psi}$ for a model $f_\psi$ is, by definition, the minimizer of $D(f_\psi, \tilde{f})$ over $\psi \in \Psi$, namely $\widehat{\psi} = \arg\min_{\psi\in\Psi} D(f_\psi, \tilde{f})$. Here, we assume the existence and uniqueness of the pseudo-true value $\tilde{\psi} \in \Psi$ and that that $\int_{-\pi,\pi} \tilde{f} d\lambda$ is finite. Through the use of Kolmogorov's formula, we apply

$$\int_{-\pi}^{\pi} \log f_\psi(\lambda) d\lambda = \log \sigma^2$$

Letting $\widetilde{f}_W$ denote the true (unknown) spectral density, the pseudo-true values are those parameter values $\widetilde{\psi}$ that minimize the KL distance from $\mathcal{F}$ to $\widetilde{f}$, i.e., $\widetilde{\psi}$ is the minimizer of $D(f_{W,\psi}, \widetilde{f})$ over $\psi \in \Psi$.

Once the pseudo-true parameter values have been computed from the minimization of the KL distance function, the power quantity defined as

$$\frac{\frac{1}{2\pi}\int_{-\pi}^{\pi} g_{\widetilde{\psi}}(\lambda)(f_{W,\widetilde{\psi}}(\lambda) - \tilde{f}(\lambda))d\lambda}{\sqrt{V_0(\widetilde{\psi})}}. \tag{9}$$

can be computed. The power quantity 9 is the major determining factor for the power of the gof signal extraction diagnostic test, being large whenever large discrepancies occur between $f_{W,\widetilde{\psi}}$ and $\tilde{f}$ at frequencies where the function $g_{\widetilde{\psi}}$ assigns greater weight. This quantity is also sometimes defined as the efficacy measure, and are labeled as *Efficacies* in the pseudo-true model panel of uSimX13.

In the pseudo-true model estimation interface of the uSimX13 software, these power quantities are computed for a given fixed stipulated SARIMA data generating process for the given data $Y_t$, and a choice of a non-nested SARIMA model. First, the pseudo-true parameters $\widetilde{\psi}$ for the proposed model $f_{W,\psi}$ are estimated using the KL distance function, then the power quantity 9 is computed for all combinations of signal-noise components (seasonal $S_t$, trend $T_t$, irregular $I_t$, and trend-irregular $T_t + I_t$). One can also compute the power quantity at any lag $h$ up to 24 by multiplying the weighting function $g(\lambda)$ by $\cos(h\lambda)$. This serves at evaluating the autocovariance structure of the signal extraction diagnostics at both nonseasonal and seasonal lags, where $h = 1$ is typically used for the trend autocovariance, and $h = 12$ and $h = 24$ for the seasonal autocovariance.

As a learning tool when performing model comparisons, the power quantity can be used to assess how close alternative models are in the sense of the KL divergence function to a posited SARIMA model. Simulating data from a SARIMA model with spectral density $\tilde{f}$, one can compare alternative models to see how close each alternative model is to the SARIMA model. As one changes parameters in the data generating SARIMA model, the effects of the power quantity and the estimated target 'pseudo-true' model can be seen along with the various spectral densities derived from $f_{W,\widetilde{\psi}}$. Using a combination of the power quantities with pseudo-true parameter values and the gof diagnostics evaluated using the pseudo-true values for an automatic model selection process is currently being investigated using the uSimX13 software.

# 5 Sliding Windows and Time Series Data Sweeping

A useful tool in modeling economic time series is the ability to compute models in subsets of the data and test for the robustness of the signal extraction and forecasting relative to a growing subset of the data. For instance, for a time series of length 300, one could estimate a model on a shorter subset of the data, say for the first 200 observations, and then increase the amount of observations, re-estimate, and then see how the model parameter values change as the number of observations or data subset increases. One can also see how the signal extractions and forecasts change with additional data. Ideally, if the model is specified correctly for the data, there should be a very small variance in the estimated parameters as more data is added to the time series. It signifies the stability of the model selection. Normally, such an exercise would be tedious to carry out with X-13ARIMA-SEATS, or any other software such as MATLAB or R as scripts or spec files would have to be written for each individual re-estimation and then re-plotted. In uSimX13 however, this task has been rendered an easy one with the addition of a sliding windows tool.

To access the sliding windows tool, the uSimX13 computation engine must be turned on, and the full length time series data loaded into the module. To turn on the sliding windows, click on the "Sliding Window Activate" check box in the main uSimX13 menu. Once clicked, the entire plotting canvas will turn to a dark shade of blue, which indicates the windowed region. To control the sliding window, place the mouse cursor along one of the edges of the canvas and *slowly* glide the mouse with the left-mouse button held down either left or right, depending on which edge of the plot canvas you are on. Moving to the left or right with the left mouse button held down, the windowed area will shrink or expand. The model parameters are estimated instantaneously as the window adjusts and in effect, all the available model statistics, diagnostics, signals, and forecasts are computed as well. For example, as the window expands or shrinks, the trend, seasonally adjusted data, and 24-step ahead forecasts can be plotted and viewed in real-time as the window changes. One can also slide the window to the left or right by placing the mouse anywhere inside the blue-windowed region, holding down the left mouse button and moving along the time domain. This way, the window length will remain fixed, but the window center will move along different subsets of the data. This can be useful for seeing how model parameters can change within regions of data that exhibit regime changes, namely a sequence in the series that suddenly changes in seasonal or cyclical structure after a certain time observation. The data can now be modeled in both sections before and after the regime change occurs in order to compare the estimated parameter values.

## 5.1 Data Sweep

With the ability to seamlessly capture partitions of the data and model within the given partition using the sliding window, a natural extension of this mouse-on-canvas utility is employ it somehow in comparing different model of the time series data. We call this method time series data sweeping

and it involves selecting an initial window of data from the first observation to the $n$-th observation where n is some number much less than the total number of observations $N$ in the data set (say, one third the amount). The data sweep then computes the sliding window from $n$ as the final observation all the way to $N$, in increments of 1. At each addition to the length of the window, the forecast is computed for up to 24 steps ahead. Of course, since the true time series data is known in the out-of-sample region of computation, we can compute the forecast error for up to $h \leq 24$ steps ahead and sum up these errors as $n$ increases to $N$. We can do this data sweep for several models, computing aggregate forecast error over time. The idea is that the best model for the data will ideally have the smallest forecast error, and thus comparing this forecast error with several models will identify the model with the best overall forecasting ability.

# 6 Conclusion

The uses and capabilities of the uSimX13 environment provides a unique time series analysis environment for both the practitioner in time series and a learning environment for time series computation. Since over half of the numerical routines and computational core comes from X-13A-S, the uSimX13 software also furnishes an attractive supplement to the X-13A-S software for learning and understanding the dynamics of time series modeling, model comparison, and goodness-of-fit diagnostics from SARIMA model estimation.

In this document we summarized most of the features of uSimX13, outlining the computational core for producing model estimates, forecasts, signal extraction components, and diagnostics. Improvements and additional modeling computational tools will consistently be added to uSimX13 to improve modeling infrastructure, improve access to data files for uploading time series data, and also make adjustments to the software to suit the needs of practitioners.

# References


[1] Akaike, H. (1973). *Information theory and an extension of the likelihood principle.* In B. Petrov and F. Czaki

(Eds.), Second International Symposium on Information Theory, pp. 267-287. Budapest: Akademia Kiado.

[2] Bell, W. R. and S. C. Hillmer (1983). *Modeling time series with calendar variation.* Journal of the American

Statistical Association 78, 526-534.

[3] Blakely, C., (2012) *iMetrica-Data Simulator: A graphical user-interface for the uploading, Simulating, and analyzing multidimensional time series.*

[4] *iMetrica-SSM: A state space modeling graphical user-interfaced environment featuring regComponent.*

[5] *iMetrica-BayesCronos: A Bayesian econometric interactive graphical user-interfaced environment for the estimation of univariate and multivariate time series.*

[6] Blakely, C., McElroy, T. *An Empirical Evaluation of Signal Extraction Goodness-of-fit Diagnostic Tests* Econometric Review (under second revision)

[7] Box, G. E. P. and G. M. Jenkins (1976). *Time Series Analysis: Forecasting and Control* (2nd ed.). San Francisco, CA: Holden-Day.

[8] Brockwell, P. and Davis, R. (1991) *Time Series: Theory and Methods* 2nd Ed. New York: Springer

[9] Dahlhaus, R. and Wefelmeyer, W. (1996) *Asymptotically optimal estimation in misspecified time series models.* Annals of Statistics **24**, 952–974.

[10] Findley, D., McElroy, T., and Wills, K. (2004) Modifications of SEATS' diagnostics for detecting over- or underestimation of seasonal adjustment decomposition components. *2004 Proceedings American Statistical Association [CD-ROM]: Alexandria, VA.*

[11] Findley, D. F., Monsell, B. C., Bell, W. R., Otto, M. C. and Chen, B. C. (1998) New capabilities and methods of the X-12-ARIMA seasonal adjustment program. *Journal of Business and Economic Statistics* **16**, 127–177 (with discussion).

[12] Golub, G. and Van Loan, C. (1996) *Matrix Computations*, Baltimore: Johns Hopkins University Press.

[13] More, J. J., B. S. Garbow, and K. E. Hillstrom (1980). *User guide for MINPACK-1.* Report ANL-80-74,

Argonne National Laboratory, Argonne, Illinois.

[14] Gomez, V. and Maravall, A. (1996), Programs TRAMO and SEATS : Instructions for the User (Beta Version: June 1997), Banco de Espana, Servicio de Estudios, DT 9628. (Updates and additional documentation can be found at http://www.bde.es/webbde/es/ secciones/servicio/software/econom.html.).

[15] Gomez, V. and Maravall, A. (2001a). Automatic modeling methods for univariate series. In D. Pena, G. C. Tiao, and R. S. Tsay (Eds.), A Course in Time Series Analysis. New York, NY: J. Wiley and Sons.

[16] Gomez, V. and Maravall, A. (2001b). Seasonal adjustment and signal extraction in economic time series. In D. Pena, G. C. Tiao, and R. S. Tsay (Eds.), A Course in Time Series Analysis. New York, NY: J. Wiley and Sons.

[17] Hannan, E. J. and J. Rissanen (1982). *Recursive estimation of mixed autoregressive-moving average models.* Biometrika 66, 265-270.

[18] Hillmer, S. and Tiao, G. (1982) An ARIMA-model-based approach to seasonal adjustment. *Journal of the American Statistical Association* **77**, 63–70.

[19] Hood, C. C. H. and Monsell, B. C. (2002) *Getting Started with X-12-ARIMA Input Files on Your PC* U. S. Census Bureau, Washington, DC

[20] Hurvich, C. M. and C. Tsai (1989). *Regression and time series model selection in small samples.* Biometrika 76, 297-307

[21] `http://www.oracle.com/us/technologies/java/overview/index.html`

[22] Koopman, S., Shephard, N., and Doornik, J. (1999) Statistical algorithms for models in state space using SsfPack 2.2. *Econometrics Journal* **2**, 113 - 166.

[23] Ljung, G. and Box, G. (1978) On a measure of lack of fit in time series models. *Biometrika* **65**, 297–303.

[24] Maravall, A. (2003) A Class of Diagnostics in the ARIMA-Model-Based Decomposition of a Time Series. Memorandum, Bank of Spain.
http://www.bde.es/servicio/software/tramo/diagnosticsamb.pdf

[25] Maravall, A. and Caporello, G. (2004) Program TSW: Revised Reference Manual. *Working Paper 2004, Research Department, Bank of Spain.* http://www.bde.es

[26] Maravall, A. (1995), Unobserved Components in Economic Time Series, in The Handbook of Applied Econometrics, eds. Pesaran, M. H. and Wickens, M., Basil Blackwell.

[27] McElroy, T. (2008) Statistical Properties of Model-Based Signal Extraction Diagnostic Tests. *Communications in Statistics, Theory and Methods* **37**, 591–616.

[28] McElroy, T. and Holan, S. (2009) *A Local Spectral Approach for Assessing Time Series Model Misspecification.* Journal of Multivariate Analysis **100**, 604–621.

[29] Monsell, B. (2007), *The X-13A-S Seasonal Adjustment Program* Proceedings of the 2007 Federal Committee On Statistical Methodology Research Conference, [Online]. Available: http://www.fcsm.gov/07papers/Monsell.II-B.pdf.

[30] Otto, M. C., W. R. Bell, and J. P. Burman (1987). *An iterative GLS approach to maximum likelihood estimation of regression models with ARIMA errors* Proceedings of the American Statistical Association, Business and Economic Statistics Section, 632-637.

[31] Schwarz, G. (1978). *Estimating the dimension of a model*. Annals of Statistics 6, 461-464

[32] Taniguchi, M. and Kakizawa, Y. (2000) *Asymptotic Theory of Statistical Inference for Time Series*. New York City, New York: Springer-Verlag.

[33] 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)