A software package for time series analysis

Adrian HOSPODAR
Training Division, RomTelecom, ahospoda@excite.com

Abstract: *This paper introduces a software package dedicated to time series analysis named ECONO. Within this presentation the following topics are covered: First, the reasons that originally motivated the development of ECONO are stated. Secondly, ECONO's main features will be briefly explained. Thirdly, the current stage of ECONO's development will be presented. Fourthly, I will present the ECONO's planned evolution. Some selected references are also included.*
Keywords: *Time series analysis, software package, nonlinear models, Bayesian analysis.*

Introduction

Time series analysis developed spectacularly during the past two decades. This is true in terms of:

- the number of techniques, methods and models aroused in this period of time
- the increasing number of time series analysis applications. Econometrics, engineering, astrophysics, psychology are just a few examples.
- the increasing diversity of theoretical background that lies behind the methods and models developed

The evolution of software tools dedicated to time series analysis has to keep the pace with this evolution. Since ECONO is and will be freeware, I will not take into the discussion the commercial software packages but only the freeware ones.

Freeware software for time series analysis can be divided into three main categories:

- freeware delivered as libraries of functions for time series analysis. Usually the functions contained in one package deal with a very specific area of time series analysis, such as Fast Fourier Transform. There are several programming languages commonly used for developing this category of freeware: Fortran, C/C++, Matlab and even Basic. Example of such libraries are:
- the STARPAC, acronym standing for the Standards Time series And Regression Package, developed at the Center for Computing and Applied Mathematics of the National Institute of Standards and Technology (formerly known as the National Bureau of Standards). The programming language is Fortran.
- FFTPACK, acronym standing for Fast Fourier Transform PACKage, developed by Paul N. Swarztrauber at the National Center for Atmospheric Research, in Boulder. The programming language is Fortran.
- ARFIT, acronym standing for AutoRegressive Fitting, developed by Tapio Schneider, AOS Program, Princeton University and Arnold Neumaier, Mathematics Institut, Wien University. The programming language is Matlab.
- freeware delivered as end-user applications. Sometimes this category of freeware attempts to integrate tools addressing different time series analysis areas. Examples in this category are:
- SANTIS, acronym standing for SignalANalysis and TIme Series processing, developed by Ralf Vandenhouten at Institute of Physiology RWTH from Aachen, Germany
- TISEAN, acronym standing for TIme SEries ANalysis, part of the output of the TISEAN project at MPI Dresden and University of Wuppertal, Germany.
- Freeware delivered as a system of libraries. I consider this to be a distinct category since it provides more than just the access to a number of functions. In my view, this category of freeware is to be considered as an emerging development environment for time series applications. The only member of this category I know

about is Ox, developed by J.A. Doornik at Nuffield College, Oxford, United Kingdom.
Considering the current state of the freeware software for time series analysis that I tried to describe above, I conclude that there is a lack of development environments for times series analysis software. My arguments are:
- the variety of programming languages used to developed the freeware from the first category presented above makes impossible their integration in a development environment.
- the end-user applications from the second category are not suitable for further development by third parties (they don't even claim to be)
- Ox, since it is the only example from the third category, can not fill the gap from what is requested and what is offered.

ECONO is developed to become a development environment for software packages dedicated to time series analysis. The main target of this project is to integrate linear and nonlinear models in a coherent manner.

2. ECONO'S main features

This section deals with the principles used in the design and implementation of ECONO
ECONO is a fully object-oriented software package. Both design and implementation are object oriented. This reduces significantly the effort of further development.
ECONO is programmed in C++. The compiler I use is Microsoft Visual C++, version 4.0, Standard Edition.
ECONO uses the development guidelines recommended by Microsoft for third part developers. For details about these development guidelines see [2]. This will ease the further development by third parties for Windows platform.
ECONO implements its functions in classes as non-member, non-friend functions. In my view this leads to a better encapsulation.
ECONO is implemented in a hierarchical system of shared libraries.
ECONO currently has two versions, named ECONO and ECONOA. While the latter is fully compatible with ANSI standards, the first makes use of several features of Windows 95 operating system. Both versions share the same source files and libraries system so the programming effort is minimized. In the rest of the paper "ECONO" is used to designate both versions. ECONOA is used to designate the ANSI compatible version only.
ECONOA is portable under Linux. I have compiled ECONOA under Linux using the Gnu C++ compiler, version 2.7.2.
ECONO provides a customizable way of inline expansion for its functions. Third party developers can use this mechanism to optimize the speed of execution and executables' size according to their needs.
ECONO can be customized to function in two modes: assisted and non-assisted. The difference between these modes is explained here in the most general manner. In both modes, ECONO is responsible for checking the coherency of users' request. When ECONO detects some incoherence in these requests it behaves differently according to the function mode. In assisted mode, ECONO informs the user about the error encountered and tries to recover from the error situation. Then it prompts the user for the right input. In the non-assisted mode ECONO supplies the missing information making use of its built-in decision rules. In the unlikely case when this is not possible, ECONO behaves just like in the assisted manner.
ECONO accepts input data stored only in double format. This is to be considered a severe limitation of the current version.
ECONO handles matrix storage and computation by making use of the NewMat library features. The NewMat version used is 09. For more details about NewMat, see [3].

3. Current stage of development

This section describes the current stage of ECONO's development. It is divided into two subsections.

The first subsection describes the ECONO's current stage of development in terms of its functionality.

3.1 The functionality of ECONO's current version

The ECONO's functions are grouped here into the following subsubsections.

3.1.1 Statistical description of data

ECONO can compute various statistical indicators. In this paper they are divided for convenience into three categories:

- Basic statistics, such as mean, variance, skewness, kurtosis.
- Statistics for time series' dynamic analyses such as autocorrelation function (ACF for short), partial autocorrelation function (PACF for short), linear impulse response function
- Statistics for measuring the nonlinear dependence of a time series such as Lyapunov exponents, correlation exponent

ECONO can also perform a variety of statistical tests useful in:

Characterizing the time series (non) stationarity

Characterizing the time series distribution

3.1.2 Transformation of time series

ECONO can perform time-series transformation in two ways:

- at the user request. A variety of built-in functions are available to the user. The user can also specify a custom function to be used in time series transformation.
- Using predefined methods in order to achieve some 'good' properties of the time series such as stationarity.

3.1.3 Time series modeling

ECONO can fit any of the following theoretical models to a time series:

- Auto Regressive Integrated Moving Average (ARIMA for short). ECONO provides several algorithms for ARIMA models estimation. All these methods make use of the Levenberg-Marqurdt (LM for short) algorithm. The LM algorithm is the standard nonlinear least squares procedure that ECONO uses.
- The ARIMA model extended to incorporate exogenous interventions.
- Auto Regressive Fractionally Integrated Moving Average (ARFIMA for short). ECONO uses a variant of the Chung algorithm for ARFIMA parameters identification.
- Threshold Auto Regressive (TAR for short). ECONO offers two methods for TAR parameters estimation:
- The graphical procedure of Tsay that is interactive (the user is prompted to choose some of the model's parameters based on a graphical representations)
- The algorithm based on the QR decomposition
- Smooth Transition Auto Regressive(STAR for short). For STAR models' parameter ECONO uses once again the LM algorithm
- Embedded Delay Threshold Auto Regressive (EDTAR for short). ECONO uses the Chen-Lee algorithm for EDTAR models' parameters identification
- Auto Regressive Conditional Heteroskedasticty (ARCH for short) family of models consisting of: ARCH, GeneralisedARCH (GARCH for short), LogarithmicGARCH (LGARCH for short) and ExponentialGARCH (EGARCH for short). ECONO provides two different methods of estimation for the GARCH models: one that is just a wrapper over the method provided by NewMat and one that is an original implementation.

3.1.4 Model identification

For ARIMA models ECONO uses two different methods for model identification. These methods are described below:

- the first method consists of fitting the ACF and PACF values to predefined theoretical models (exponential, sinusoidal and combination of these two). Based on some decision rules ECONO chooses the ARIMA model orders that best describes the times series dynamic.
- The second method uses informational criteria (IC for short) for model identifica-

tion. ECONO allows the user to choose between the AkaikeIC (AIC for short), the BayesianIC (BIC for short) and the Schwarz IC (sometimes also called phi IC). The default information criteria ECONO uses is BIC.
For TAR and EDTAR models ECONO uses the following information criteria: BIC, Generalised IC and informational complexity criterion.
ARMA, TAR and EDTAR models are the only ones for which ECONO provides model identification tools. This situation is not expected to change in the next release.

3.1.5 Bayesian Analysis
I consider Bayesian analysis in the time series context to be a coherent framework. ECONO uses the Bayesian Model Averaging technique to build mixed models for time series analysis. The following models are used for the BMA: ARIMA, ARFIMA, TAR and EDTAR.
3.2 The hierarchy of classes in ECONO current version
ECONO current version consists of the following classes:
CTmeS. This is ECONO's base class. It is responsible for storing and handling time series data. It computes the statistics listed in the subsubsection 3.1.1 Statistical Description of Data. It implements basic time series transformations.
CData. This class is the interface between CTmeS and data files. It is responsible for data reading.
CRezid. This class implements the residuals' analysis methods. It also offers goodness of fit diagnosis for all the models listed in the subsubsection 3.1.3 Time Series Modeling.
CArma. Implements ARIMA model estimation of parameters, ARIMA model selection, goodness of fit techniques for ARIMA model. It also implements time series transformations such as detection and removal of outliers, detection and removal of (non) permanent mean level modifications.
CArch. Implements ARCH model estimation of parameters and goodness of fit techniques.
CArfima. Specializes CArma class for ARFIMA models offering the same functionality as its base class.
CTar. Specializes CArma class for TAR models offering the same functionality as its base class.
CStar. Specializes CArma class for STAR models offering the same functionality as its base class.
CEdtar. Specializes CTar class for EDTAR models. Only some of its base class' functionality is preserved (it does not offers model identification, for example).
CGarch. Specializes CArch class for GARCH models offering the same functionality as its base class.
CEgarch. Specializes CGarch class for EGARCH models offering the same functionality as its base class.
CLgarch. Specializes CGarch class for LGARCH models offering the same functionality as its base class.

4. ECONO's planned evolution

For ECONO, the targeted users are the members of the academic community. To ease making the decision if ECONO interests you, I will present the planned evolution of ECONO:
First of all, ECONO is freeware and will remain freeware. Third party developers are encouraged to add functionality to this project.
ECONO current version has no graphical capabilities. ECONO's next release will surely include a graphical library. It is likely that this graphical library will be GnuPlot, version 3.7.
The matrix library that ECONO uses will be changed in the near future (perhaps in the next release). The current matrix library, NewMat 09, is inefficient in handling matrices with small numbers of rows. This is often the case with time series analysis. The NewMat functionality is also to be considered reduced.

It is likely that in the near future ECONO will be able to deal with data stored in the following formats: double, integer and a custom extended precision format.
ECONO will implement, starting with the next release, methods for sparse matrices identification. This will lead to an increase in terms of speed of execution.
The number of supported file formats will be increased in the next release. I will keep increasing this number from version to version. The next release will allow the user to access data stored in Excel worksheets, FoxPro and Access databases and SANTIS native format data files. It is likely that there will be an ECONO native file format in the next release.
ECONO will keep in the following releases to implement sections of code critical for speed of execution in code-machine. The current version does this in approximately 60% of the cases where this is needed. I expect that this figure will increase in the next release at about 70%.
The next release of ECONO will have an interface with the Ox package. This will allow the users to use both ECONO and Ox functions either from Ox or ECONO.
On long term, ECONO will extend model identification mechanisms for the following models: ARFIMA, ARCH and GARCH.
ECONO will implement, starting with the next release, random number generators. The current version uses the NewRan library of random numbers, version 2A.

Acknowledgements
The author thanks to Mr. Ivan Ion from the Economics Informatics Department, Academy of Economic Studies, Bucharest, Romania.
The author wants also to manifest his gratitude to the Economic Researches Department of the Academy of Economic Studies for supporting the development of ECONO.

References
1. Terence C. Mills, Time series techniques for economists, Cambridge University Press, 1990
2. Jocelyn Gracer, Microsoft Foundation Class Library Development Guidelines, Visual C++ Users Guide, Microsoft Press, 1995
3. R.B. Davies, Documentation for NewMat 09, a matrix library in C++, http://webnz.com/robert/, 1998
4. P. Sinioros, Software reliability monitoring a quality characteristic during the whole life cycle, in Computer Science, Bucharest 1997