OxMetrics news | AUGUST 2005 ISSUE 5

Announcing the release of OxMetrics™ 4

OxMetrics™ is a modular software system for econometric and statistical analysis. Some of the OxMetrics™ modules started as independent programs but are now available as modules of OxMetrics™. The current modules are: Ox Professional™, PcGive™, STAMP™, PcGets™, PcNaive™, G@RCH™ and TSP/GiveWin™.

All modules are described in the Website http://www.oxmetrics.net.

With the release of version 4, Timberlake Consultants (the distributors of OxMetrics™) will be offering a new pricing structure to help OxMetrics™ users to acquire all the modules.

Introducing the next generation of OxMetrics™

by Jurgen A. Doornik

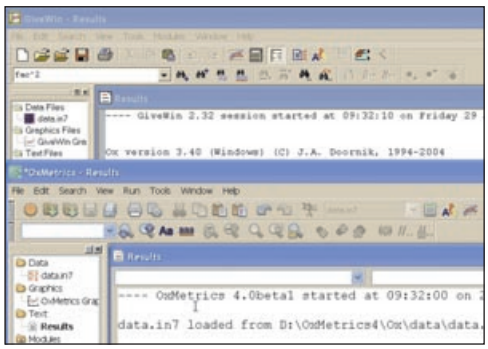
The OxMetrics™ software has undergone a complete interface overhaul. This is a major operation, which explains why it has been a while since the last full upgrade (although there has been a stream of point upgrades introducing new functionality). It is the third such overhaul in about fifteen years that I am involved with. The first was the introduction of a text-based menu and dialog system that could be operated with a mouse - this was around 1990, when most systems were running MS-DOS as the operating system. The next step was towards 32-bit Windows, giving a much more advanced interface, and the ability to have text and graphics windows side by side. We also introduced the modular structure that survives to this day: GiveWin was the front-end to which econometric modules such as PcGive, STAMP, TSP, etc. would connect. GiveWin provides the data, and, after model formulation and estimation, receives back the text and graphs.

The modular structure is maintained in the new version, because it has proven to be a major advantage of the OxMetrics system: components can be developed entirely independently, while being presented within a unified framework. This is confirmed by the new modules that have recently become available: PcGets and G@RCH, together with the increasing number of downloadable OxPack applications.

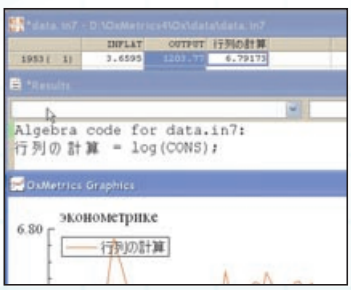
The new version of the software is written using a different interface library (wxWidgets, see www.wxwidgets.org). We also decided to drop the `GiveWin' name (which I was never fond of) in favour of `OxMetrics'. The main aims of the transition are:

Better support for Windows XP™
Of course, the current generation runs under Windows XP without problems, but it doesn't look quite right. The new version will have the XP look and feel as expected. In the following screen capture, the old GiveWin is on top, with the new OxMetrics below:

Support for most languages
Internally OxMetrics and OxEdit are now fully Unicode, which means that languages with complex character sets, such as Chinese and



Japanese are supported: you can enter Japanese text, have Chinese variable names and Russian text in your graphs:



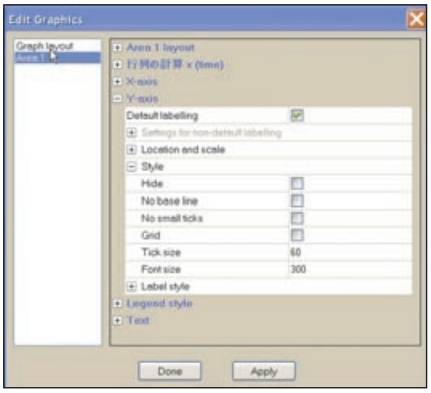
Support for most platforms
Ox already runs on many platforms, and we always wanted the rest of our software to do that too, which is why we adopted wxWidgets. The intitial release will be Windows only, because this is what the majority of users work with (not surprisingly, since that was all that GiveWin supported). But we do have Linux and Apple Mac™ computers running to develop OxMetrics on, and Linux will be the next target.

This is the calculator in action on a Mac:

Focussing on the OxMetrics front-end, there are several new features that make the program easier to use:

Improved Dialogs
We have tried to make dialogs clearer and easier. In particular, all dialogs can be enlarged if you wish to have more space to enter the required information.

Dialogs are also more `linear'. In some cases (graph editing is an example), we fell in the trap of cramming too many controls in both horizontal and vertical groupings. Now options are listed more simply, making items easier to find. For example, when double-clicking on an axis you'll see the new dialog:



Undo/Redo
Text already had unlimited Undo and Redo facility. Now database and graph changes can also be undone.

Modular Structure
As mentioned above, the modular structure is fully maintained. However, now we offer a logical way of selecting a modelling action, rather than a module based choice. For example, in the old structure you would choose between STAMP and PcGive (say), then with PcGive you had to choose a package, and then within the package a model class (such as static versus dynamic panel models). This reflects the structure as implemented by the developers. In the new version you first choose a model category, for example, Time-series Models or Cross-section Models. Then, within the category a model type: ARFIMA or Seasonal Adjustment for the former, Cross-section Regression or Logit Models for the latter. OxMetrics will work out whether the model is part of STAMP or PcGive, and switch to the relevant module for you.

I presented the major changes in the new generation of the OxMetrics front-end. There are many smaller improvements which will become clear when using the new version. Together they make the program easier to use and more powerful at the same time.

The new STAMP™ 7

by Siem Jan Koopman

The STAMP program enters its seventh version. The current version 6 of STAMP is based on univariate and multivariate versions of the unobserved components time series model. It can handle model-based decompositions of a time series or a panel of time series based on trend, cycle, seasonal, autoregression and irregular components. Further, the model in STAMP can include regression variables and interventions for outlier, trend and slope breaks. Apart from the unobserved components modeling framework, STAMP is also unique because it is based on up-to-date state space technology. Although the algorithms associated with state space are usually regarded as complex, the practical use of the STAMP program is surprisingly user-friendly. The STAMP user only need to concentrate on the practical aspects of analysing, modeling and forecasting of time series. The computationally efficient implementations of Kalman filter and smoothing computations are done in the background of STAMP. Although the statistical output can be extensive, the default is to output a limited number of diagnostics. Many graphical options are available to gain insights in the successes or failures of the modeling strategy for a particular time series. The estimated decomposition of trend, cycle, seasonal and irregular can be viewed instantaneously in a single matrix of graphs.

Since we regard STAMP 6 as being a success, the user may wonder why we are developing a new version. The main reason is that more applications of the state space methodology have been developed in statistics, economics and finance. For example, state space applications for business cycle modeling, seasonal adjustment, forecasting, volatility modeling, etc. have appeared in the literature and are in demand. Although this wide variety of applications have the state space framework in common, the nature of the applications is quite distinct. Therefore, we have decided to develop a STAMP for the future, a STAMP that is build as a collection of modules. In the next release of STAMP 7, four modules will be present:

- Unobserved component models
- Business Cycles
- Seasonal adjustment
- Forecasting

In future, we plan to have other modules for Stochastic Volatility models, Bayesian dynamic linear models, Count data, SsfPack Interactive, etc.

The module Unobserved components is similar to the current STAMP 6 but has the following new features: (i) wider variety of components to select, (ii) more flexible handling of parameter estimation, (iii) treatment of missing values, (iv) more test-statistics and diagnostics with p-values, (v) more flexible options for component graphics, all with confidence intervals, (vi) weight functions or kernels, (vii) frequency gains, and much more. An example of the new graphical output in STAMP is presented in Figure 1.

The module Business Cycle offers both nonparametric filters such as Hodrick-Prescott, Baxter-King, Christiano-Fernandes and model-based approaches such as the ones of Beveridge-Nelson, Clark and Harvey-Jaeger. Appropriate output is presented for the estimated business cycle indicator, weight functions, frequency gains, turning points, revisions, etc. Multiple series can be selected and multiple business cycle methods can be selected to facilitate comparison and production. An example of the graphical output from this module is presented in Figure 2.

The module Seasonal adjustment contains basic nonparametric seasonal adjustments but

also model-based treatments using unobserved component models and a selection of seasonal ARIMA models. The output consists of standard diagnostics for seasonal adjustment based on time-domain and frequency-domain statistics. Some standard output of the Bureau of the Census X12 program is presented as well. Multiple series can be selected for a fully automatic seasonal adjustment.

The module Forecasting presents a selection of ad-hoc forecasting methods that are implemented in a state space framework so that standard errors of the forecasts can be generated. The usual forecast criteria can be requested and multiple series can be selected for automatic forecasting. Also, the user can consider model-based forecasting on the basis of unobserved component models, ARIMA models and "single-source-of-error" models.

The module-wise functionality of STAMP 7 reflects the wide applicability of state space methods in statistics, economics and finance. Further, it is anticipated that the separate modules can be maintained in a more sizeable form and therefore it will be easier to keep STAMP 7 up-to-date and bug-free!

Finally, STAMP 7 is part of the new Oxmetrics 4 generation. This implies that the new database, text and graphical utilities become available to the STAMP users and that STAMP will be multi-platform in future.

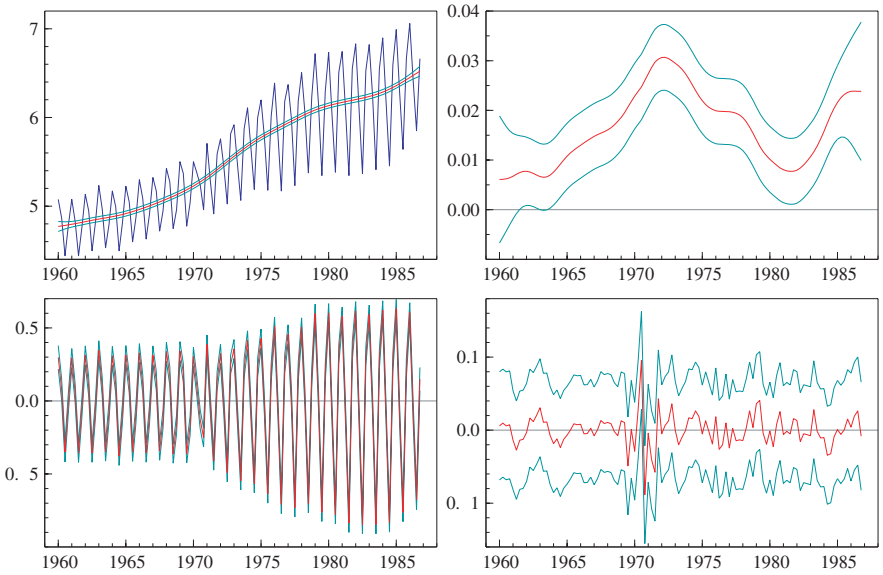


Figure1: Clock-wise starting from the upper-left: Time series with estimated trend, estimated slope, estimated seasonal and estimated irregular. Estimated components are presented with 95% confidence intervals.

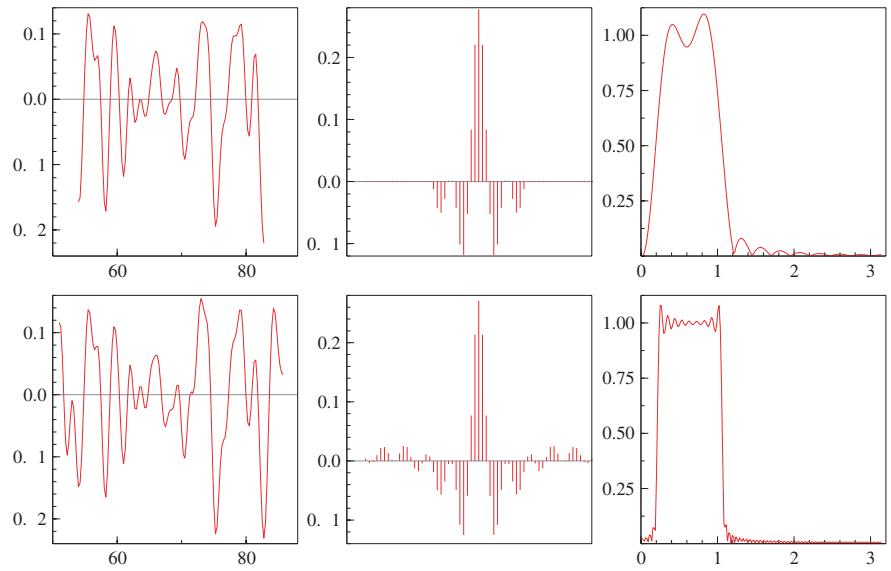


Figure2: Row-wise: Baxter-King and Christiano-Fitzgerald filters; Column-wise: the extracted business cycle, the weight or kernel function and the spectral gain function.

Two students' views of G@RCH™

The post-graduate students at Cass Business School use OxMetrics heavily during their projects. They were the first students in the UK to use the G@RCH module. The MSc students provided us with the following assessment

G@RCH Package

During my course MSc In Quantitative Finance I used the OxMetrics modules PcGive, and more G@RCH, in many occasions.

During the course Advanced Econometric Analysis of Financial Markets2, We used G@RCH to model the conditional variance of financial time series. The variety of options and settings provided enabled me to understand the different models of the ARCH family and "experiment" with them in order to identify which model fits particular time series the best. In addition, G@RCH provides simultaneous estimation of the models of conditional mean and conditional variance and this proved to be more accurate than results provided by other software available at CASS.

I also used G@RCH to calculate financial risk analysis, and more specifically, in the estimation and calculation of Value at Risk. In particular, G@RCH was very useful in forecasting VaR of assets and portfolios.

Finally, during my dissertation on the impact of macroeconomic announcements on asset pricing, I used G@RCH and PcGets. After estimating the econometric model using PcGets and conducting the misspecification tests, G@RCH helps improving my model and estimate the conditional variance model of the time series and capture the heteroskedasticity of the residuals.

Conclusively, the G@RCH package has provided the maximum contribution to my econometric courseworks and research, and I will definitely continue to use it.
Konstantina Founta

During the advanced financial econometrics course, I used G@RCH to model and forecast the conditional variance in a data generating process. Particularly, the wide range of options available in this package makes it suitable towards complex econometric problems.

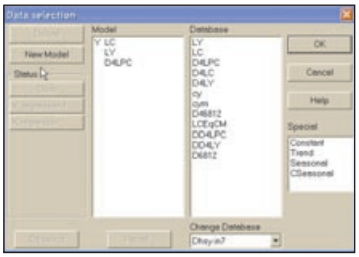
The first time I used G@RCH package was for my coursework which included modelling and forecasting fractionally integrated processes. I had to model hundreds of equations using G@RCH to fit an appropriate data generating process. G@RCH provides ample of options in order to estimate the model of your choice. For instance, when modelling the return series of currency markets that showed fat tails, G@RCH provided an options for fat tailed distribution in estimation.

I found the G@RCH manual very useful when I started using the software. I have used G@RCH for many other applications and also find it particularly useful for conducting descriptive statistics, misspecification tests and forecasting.
Dennis Philip

Quick Modeller in PcGets™

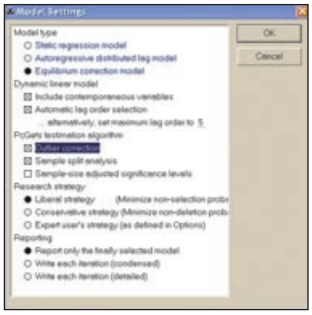
by David F. Hendry and Hans-Martin Krolzig

To extend the practical realm of automatic selection in PcGets, we have developed a mode for a non-expert user[1]. The user simply specifies the appropriate functions of the regressand and the basic regressors, then PcGets creates the GUM and selects a model. Thus, the input to `quick modeller' is just the list of `basic variables', y_t, $x_{1,t}$, …, $x_{N,t}$. A constant is automatically included, but can be eliminated. The first screen capture illustrates:



The next dialog offers three model types:
1. static regression model;
2. an autoregressive distributed lag model; and
3. an equilibrium-correction model.

(1. This is briefly discussed in Hendry and Krolzig (2005), `The properties of automatic Gets modelling,' Economic Journal}, 115 , C32--C61)



Depending on which type is selected, the dialog either moves directly to model selection (case 1.) or the user sets a small series of options about the dynamic model (case 2. and 3.) concerning the inclusion of contemporaneous variables or not, and setting the maximum lag length or allowing automatic lag selection as the second screen capture shows. The maximum lag length is set from the data frequency, given the sample size.

Then there are options about the algorithm, such as outlier correction; analyzing a sample split, and adjusting the significance levels for the size of the sample. The research strategy also can be selected from the usual three:
a) `Liberal';
b) `Conservative'; and
c) `Expert'.

The first of these is the default, and corresponds to roughly a 5% null rejection frequency per test, whereas the `Conservative' is closer to 1%. The expert allows the user to set all of the many significance levels, which is probably inadvisable for non-experts! Finally, the detail of the level of reporting can be chosen. In each case, a default is set such that the user does not need to make any choices. Once that dialog has been accepted, only the sample size remains to be set, with full sample being the default. Acceptance leads to the `testimation algorithm' commencing.

First, the levels equation is estimated unrestrictedly; and the congruence of the resulting GUM is checked. Seasonality, and a possibly inappropriate lag selection, are considered. Next, the *PcGive* unit-root test is computed, and the variables transformed to differences and any cointegration combination. The order of differencing is based on the largest lag coefficient, often the first, to facilitate selection of a parsimonious model. Finally, that I(0) representation is re-estimated, and the usual procedures for selecting a parsimonious, congruent, undominated model re implemented.

As an example we take the data set from DHSY, and just input UK consumers' expenditure variables c, y, and $\Delta_4 p$ with a maximum lag selected as 5 (as the data are quarterly and not seasonally adjusted) using the Liberal strategy with outlier correction.

A summary of the output is: See Figure A below.

$$\widehat{\Delta_4 c_t} = \underset{(0.013)}{-0.104}\, e_{t-4} + \underset{(0.036)}{0.301}\, \Delta_4 y_t + \underset{(0.036)}{0.169}\, \Delta_4 y_{t-1} - \underset{(0.034)}{0.192}\, \Delta_4 (\Delta_4 p)_t \qquad (1)$$

$$R^2 = 0.84 \;\; \widehat{\sigma} = 0.67\% \;\; F_{\mathsf{Chow}(1974{:}4)}(7,58) = 0.52 \;\; F_{\mathsf{ar}(1-4)}(4,61) = 1.03$$

$$\chi^2_{\mathsf{nd}}(2) = 0.83 \;\; F_{\mathsf{arch}(1-4)}(4,61) = 1.16 \;\; F_{\mathsf{het}}(8,60) = 1.09 \;\; T = 1959(2) - 1976(2)$$

where the long-run equilibrium is:

$$e = c - 0.998y + 1.138\Delta_4 p. \qquad (2)$$

Thus, 'quick modeller' selects the equation reported by DHSY without their dummy variable — and in seconds as against their large modelling time input!

Figure A

New Random Number Generators in Ox

by Jurgen A. Doornik

There has been much research in recent years into very-high period random number generators (RNGs). The emphasis has been on generators with desirable mathematical properties, good output when tested empirically, and high efficiency in terms of computational speed[1].

Ox has one high-period RNG built-in due to Pierre L'Ecuyer and with period 2^113. However, the default is still the old-fashioned linear-congruential generator by Park and Miller. This is set to change with Ox version 4.

(1. See my chapter in the forthcoming Palgrave Handbook of Econometrics for more details and references)

New: very high period

The new default random number generator uses a multiply-with-carry algorithm and is due to George Marsaglia. It has a period of 2^8222. With current computers it takes about half a minute to draw all 2^31 numbers provided by the Park and Miller RNG. Drawing `only' 2^64 takes more than 4000 years, so 2^8222 should be plenty, even when sliced up for parallel use.

New: very high resolution

More novel is the provision of a high-resolution generator. Ox computes with so-called `doubles', which are 64 bits in size. Of that, 52 bits are for the mantissa, while the rest is for the sign and exponent. However, almost all RNGs that are currently in use (including those of Ox 3) provide only 32 random bits, with the remaining twenty set to zero. In decimal terms, the equivalent is that the double allows for about 15 significant digits, but the last 5 are always zero (because it happens at the bit level, the zeros cannot be seen when printing the numbers). The Ox 4 default generator does have 52 random bits in the uniform random numbers (and by derivation in all other distributions), making Ox quite unique.

New: standard normal

Ox also uses a new algorithm to generate standard normal random numbers. This is based on the so-called Ziggurat method, an efficient method, which, however, had to be corrected for some shortcomings (see my paper at www.doornik.com/research.html). In combination with the higher resolution, standard normal random numbers now range from -11.9 to 11.9, rather than -6.23 to 6.23.

A comparison

The following table shows some benchmarks (on my dual Opteron 1.8 Ghz machine) for generating and computing the mean of 10^9 random numbers.

	uniform: ranu	**normal: rann**
Ox 3 default	27 seconds	76 seconds
Ox 3 ranseed ("LE")	38 seconds	106 seconds
Ox 4 default	33 seconds	56 seconds

It shows that the efficient implementation together with the Ziggurat algorithm more than makes up for the cost of providing high-period and high-resolution normal random numbers.

It is quite likely that these changes will not make a qualitative difference in most simulation experiments. However, there may be some extreme cases were it does, and it is reassuring that Ox advances the state of the art in this respect.

TSP/GiveWin Version 5.0

by Bronwyn Hall

TSP/GiveWin features the following enhancements:

- Completely revised User's Guide and Reference Manual, in a new compact paperback format, together with a completely revised interactive HELP system (context-based for TSP/Givewin).

New PANEL estimation commands:

- PANEL(REI) for individual Random Effects using Maximum Likelihood (ML) estimation. ML provides results which are easier to reproduce, as opposed to the many choices of variance components estimators for 2-step GLS. Automatic checking for multiple optima, t-statistics for the variance components, and direct Likelihood Ratio tests.
- PANEL(REIT) for individual and time Random Effects using ML estimation. Handles unbalanced data and large numbers of individuals.
- AR1(REI or FEI) for individual Random or Fixed Effects, via exact ML estimation. Fixed Effects also via conditional ML (NLS) estimation, which can handle irregularly spaced data.
- 2SLS(FEI), LIML(FEI), SUR/3SLS/GMM/FIML(FEI) for individual Fixed Effects. Standard errors are available for the fixed effects in 2SLS. Computed by individual mean-differencing the data and instruments prior to estimation.
- PROBIT(REI or FEI) for individual Random or Fixed Effects. The user may specify the number of Hermite quadrature points for random effects. Standard errors for Fixed Effects.

Improved graphics:

- PLOT, GRAPH(LINE) handle panel data, by inserting gaps between individuals.
- GRAPH(NOSORT,LINE) allows lines which are joined in time order, rather than being sorted by the X values. SYMBOL option puts large symbolts at each point. unix versions have graphics, using gnuplot interface and .GIF output.
- HIST graphical output (Windows/GiveWin version only).

Other new estimation procedures:

- LAD(LOWER=lowerlimit,UPPER=upperlimit) - Censored Quantile Regression. SEs/VCOV via bootstrapping; also available for uncensored LAD and quantile regression.
- TOBIT(LOWER=lowerlimit,UPPER=upperlimit) General 2-Limit Tobit.
- INTERVAL for Interval regression. This is like an ordered probit, but the bound values between the categories are known.
- KERNEL(BANDWIDTH=bandwidth,RELBAND=relativeband, IQR) x; or KERNEL y x; for Gaussian Kernel density and univariate regression.

Improvements to existing commands:

- ANALYZ extensions:
  (1) allow expressions that vary over observations (such as elasticities for several time periods)

Continued overleaf...

2) asymmetric confidence intervals, using Monte Carlo.

- BJEST(HCOV=U), ML PROC (HCOV=U) yield parameter standard errors from numeric second derivatives (new default). Usually much better than the BFGS or DFP rank 1 update methods.
- FIML(HCOV=C,HITER=C) gives parameter standard errors from Discrete Hessian (a very close approximation to analytic Hessian, based on numerical differences of the analytic first derivatives). Can also be used for iteration.
- Regression diagnostics added:
  (1) Chow test robust to heteroskedasticity.
  (2) AR1(OBJFN=GLS) - Common factor test.

READ now handles all current Excel files (2/3/4/5/7/98/2000/2002). This is especially helpful for datasets with more than 32768 rows, although there is still a limitation of 65535 rows. READ can also handle Stata v7 files.

3rd OxMetrics User Conference

17-18 August 2005

Centre for Econometric Analysis (CEA@Cass), Cass Business School, 106 Bunhill Row, London, EC1Y 8TZ (U,K.)

Conference sponsors
Cass Business School
Timberlake Consultants Ltd

Programme

17 August 2005

Session 1: PcGets (1)
Chairperson: Giovanni Urga
David F. Hendry (Oxford, UK) and Hans-Martin Krolzig (University of Kent, UK) "'Quick Modeller' and Other New Features in PcGets"
David F. Hendry (Oxford, UK). Comment on "Dummy Saturation Distributions and the Resulting Automatic Tests for Super Exogeneity"
Neil R. Ericsson (Division of International Finance, Federal Reserve Board, Washington, DC) "Constructive Data Mining: Modeling Australian Inflation"

Session 2: OxMetrics Developments
Chairperson: James Davidson
Siem Jan Koopman (Free University Amsterdam, Netherlands) "Economic and financial time series analysis using STAMP 7 and SsfPack 3"
Jurgen Doornik (Oxford University) "OxMetrics 4: The Next Generation"

Round Table Discussion with OxMetrics Developers
Following a 5-10 minute introduction each from Jurgen Doornik, David Hendry, and Siem Koopman, the main aim of the round table is to provide a forum for an exchange of suggestions and ideas for future developments of the software.

19:00 Conference dinner

18 August 2005

Session 3: Cycles and Unit Roots
Chairperson: Siem Jan Koopman
Andrew Harvey (Cambridge) "Trend and Cycles in Economic Time Series: A Bayesian Approach"
Peter Boswijk "Adaptive Testing for a Unit Root with Nonstationary Volatility"

Session 4: PcGets (2)
Chairperson: Jurgen Doornik
Jennifer Castle and David Hendry "Extending the Boundaries of PcGets: Non-linear Models"
Luc Bauwens and Genaro Sucarrat (CORE and the Department of Economics, Universite Catolique de Louvain, Belgium) "General-to-Specific Modelling of Exchange Rate Volatility: A Forecast Evaluation"

Session 5: Volatility and Long Memory
Chairperson: Marius Ooms
Jonathan Dark (Department of Econometrics and Business Statistics, Monash University, Australia) "Bivariate Error Correction FIGARCH and FIAPARCH"
Lucio Della Ratta (Cass Business School, London, London, U.K.) and Giovanni Urga (Cass Business School, London, U.K.) "Modelling Credit Spread: A Fractional Integration Approach"

Session 6: Time Series Modelling
Chairperson: Peter Boswijk
Rasoul Sajjad (AFM Department,University of Essex) "Using Markov Regime Switching Framework to Estimate Dynamic VaR in Foreign Exchange Markets"
James Davidson (Exeter) "Recent Developments in Time Series Modelling 4"

There will be a small conference fee of £84 + VAT = £98.70 to cover costs for coffee, teas, luncheons and conference dinner on the 17th of August.

Training courses

Financial Modelling and Forecasting using PcGive and PcGets, 10-12 October 2005, Milan, Italy (course given in Italian)

Financial Modelling and Forecasting using PcGive and PcGets, 14-16 December 2005, London, UK

TIMBERLAKE CONSULTANTS LTD
HEAD OFFICE: Unit B3, Broomsleigh Business Park, Worsley Bridge Road, London SE26 5BN UK
Tel.: +44 (0)20 8697 3377 | Fax: +44 (0)20 8697 3388 | e-mail: info@timberlake.co.uk
website: http://www.timberlake.co.uk
OTHER OFFICES & AGENCIES: Brazil, Poland, Portugal, Spain, and U.S.A.

Your contact: