Econometric Modelling of Markov-Switching Vector Autoregressions using MSVAR for Ox

BY HANS-MARTIN KROLZIG *
Institute of Economics and Statistics and *Nuffield College, Oxford.*

December 15, 1998

Contents

1 Introduction

MSVAR (Markov-Switching Vector Autoregressions) is a package designed for the econometric modelling of univariate and multiple time series subject to shifts in regime. It provides the statistical tools for the maximum likelihood estimation (EM algorithm) and model evaluation of Markov-Switching Vector Autoregressions as discussed in Krolzig (1997b). A variety of model specifications regarding the number of regimes, regime-dependence versus invariance of parameters etc. provides the necessary flexibility for empirical research and will be of use to econometricians intending to construct and use models of dynamic, non-linear, non-stationary or cointegrated systems.

MSVAR is a class written in Ox (see Doornik, 1998), and is used by writing small Ox programs which create and use an object of the MSVAR class. Some knowledge of Ox will be required to use MSVAR.

Ox is an object-oriented matrix language with a comprehensive mathematical and statistical function library. Matrices can be used directly in expressions, for example to multiply two matrices, or to invert a matrix. Use of the object oriented features is optional, but facilitates code re-use. The syntax of Ox is similar to the C, C++ and Java

*I benefited greatly from comments of Mike Clements, Jurgen Doornik, Juan Toro and Carolina Sierimo .

languages. This similarity is most clear in syntax items such as loops, functions, arrays and classes. The MSVAR class derives from the Database class to allow the easy use and exchange with other classes such as PcFiml.

An additional simulation class (in development) allows Monte Carlo experimentation of the facilities in the estimation class.

2 Disclaimer

This package is functional enough to be useful, but by no means finished yet (see the short to do list at the end of this paper). This package is provided as is, and you may use it at your own risk. Please report any problems or suggestions for improvement to the author (email: Hans-Martin.Krolzig@nuffield.oxford.ac.uk).

This package must be cited whenever it is used.

3 Ox version

MSVAR requires Ox version 2.00 or later. To run the program in §9.3 under Windows 95/NT:

```
oxl kroto
```

You can also use OxRun to run the program in §9.3 under Windows 3.1/95/NT. In that case the output will appear in GiveWin, instead of on the MS-DOS console. MSVAR is written as 100% pure Ox code, and will also work on Unix platforms.

4 Installation

Create a `msvar` subdirectory in the `ox\packages` directory and put `msvar.zip` in that directory and unzip `msvar.zip`[1] into that directory.

This allows for running files from that directory. MSVAR uses the `#import` statement (introduced with Ox 2.00) to allow convenient running of the package from any directory. Add `#import<packages/msvar/msvar>` at the top of your files to achieve this. You also might want to add the `msvar` subdirectory to your `OXPATH` statement.

5 Main files

- `msvar.h` – the header file for the MSVAR class;
- `msvar.oxo` – the compiled source code.
- `hmk.h` – the header file for some general functions used by the MSVAR class;
- `hmk.oxo` – the compiled source code.
- `msvar.pdf` – this document.

The remaining files are sample programs and data.

6 Data organization

The following data files can be read directly into an MSVAR object: GiveWin (.in7/.bn7), spreadsheet (Excel, Lotus), ASCII and Gauss (.dht/.dat). This is explained in the Ox manual.

[1] Available for downloading through www.economics.ox.ac.uk/hendry/krolzig

7 Markov-switching vector autoregressions

7.1 Types of regime-switching models

Reduced form vector autoregressive (VAR) models have been become the dominant research strategy in empirical macroeconomics since Sims (1980) and implemented in programs as PcFiml (see Doornik and Hendry (1997)). The MSVAR class provides tools to estimate VAR models with changes in regime.

When the system is subject to regime shifts, the parameters θ of the VAR process become time-varying. But the process might be time-invariant conditional on an unobservable regime variable s_t which indicates the regime prevailing at time t. Let M denote the number of feasible regimes, so that $s_t \in \{1, \ldots, M\}$. Then the conditional probability density of the observed time series vector y_t is given by

$$p(y_t|Y_{t-1}, s_t) = \begin{cases} f(y_t|Y_{t-1}, \theta_1) & \text{if } s_t = 1 \\ \vdots & \\ f(y_t|Y_{t-1}, \theta_M) & \text{if } s_t = M, \end{cases} \tag{1}$$

where θ_m is the VAR parameter vector in regime $m = 1, \ldots, M$ and Y_{t-1} are the observations $\{y_{t-j}\}_{j=1}^{\infty}$.

Thus, for a given regime s_t, the time series vector y_t is generated by a vector autoregressive process of order p (VAR(p) model) such that

$$\mathsf{E}[y_t|Y_{t-1}, s_t] = \nu(s_t) + \sum_{j=1}^{p} A_j(s_t) y_{t-j},$$

where $u_t = y_t - \mathsf{E}[y_t|Y_{t-1}, s_t]$ is an innovation process with a variance-covariance matrix $\Sigma(s_t)$, assumed to be Gaussian:

$$u_t \sim \mathrm{NID}(\boldsymbol{0}, \Sigma(s_t)).$$

If the VAR process is defined conditionally upon an unobservable regime as in equation (1), the description of the data generating mechanism has to be completed by assumptions regarding the regime generating process. In Markov-switching vector autoregressive (MS-VAR) models – the subject of this study – it is assumed that the regime s_t is generated by a discrete-state homogeneous Markov chain:[2]

$$\Pr(s_t|\{s_{t-j}\}_{j=1}^{\infty}, \{y_{t-j}\}_{j=1}^{\infty}) = \Pr(s_t|s_{t-1}; \rho),$$

where ρ denotes the vector of parameters of the regime generating process.

The MS-VAR model belongs to a more general class of models that characterize a non-linear data generating process as piecewise linear by restricting the process to be linear in each regime, where the regime is conditioned is unobservable, and only a discrete number of regimes are feasible. These models differ in their assumptions concerning the stochastic process generating the regime:

(i.) The *mixture of normal distributions* model is characterized by serially independently distributed regimes:

$$\Pr(s_t|\{s_{t-j}\}_{j=1}^{\infty}, \{y_{t-j}\}_{j=1}^{\infty}) = \Pr(s_t; \rho).$$

In contrast to MS-VAR models, the transition probabilities are independent of the history of the regime. Thus the conditional probability distribution of y_t is independent of s_{t-1},

$$\Pr(y_t|Y_{t-1}, s_{t-1}) = \Pr(y_t|Y_{t-1}),$$

and the conditional mean $\mathsf{E}[y_t|Y_{t-1}, s_{t-1}]$ is given by $\mathsf{E}[y_t|Y_{t-1}]$. Even so, this model can be considered as a restricted MS-VAR model where the transition matrix has rank one. Moreover, if only the intercept term will be regime-dependent, MS(M)-VAR(p) processes with Gaussian errors and $i.i.d.$ switching regimes are observationally equivalent to time-invariant VAR(p) processes with non-normal errors. Hence, the modelling with this kind of model is very limited.

[2] The notation $\Pr(\cdot)$ refers to a discrete probability measure, while $p(\cdot)$ denotes a probability density function.

(ii.) In the *self-exciting threshold autoregressive* SETAR(p, d, r) model, the regime-generating process is not assumed to be exogenous but directly linked to the lagged endogenous variable y_{t-d}.[3] For a given but unknown threshold r, the 'probability' of the unobservable regime $s_t = 1$ is given by

$$\Pr(s_t = 1 | \{s_{t-j}\}_{j=1}^{\infty}, \{y_{t-j}\}_{j=1}^{\infty}) = I(y_{t-d} \leq r) = \begin{cases} 1 & \text{if } y_{t-d} \leq r \\ 0 & \text{if } y_{t-d} > r, \end{cases}$$

While the presumptions of the SETAR and the MS-AR model seem to be quite different, the relation between both model alternatives is rather close. This is also illustrated in the appendix which gives an example showing that SETAR and MS-VAR models can be observationally equivalent.

(iii.) In the *smooth transition autoregressive* (STAR) model popularized by Granger and Teräsvirta (1993), exogenous variables are mostly employed to model the weights of the regimes, but the regime switching rule can also be dependent on the history of the observed variables, *i.e.* y_{t-d}:

$$\Pr(s_t = 1 | \{s_{t-j}\}_{j=1}^{\infty}, \{y_{t-j}\}_{j=1}^{\infty},) = F(y'_{t-d}\delta - r),$$

where $F(y'_{t-d}\delta - r)$ is a continuous function determining the weight of regime 1. For example, Teräsvirta and Anderson (1992) use the logistic distribution function in their analysis of the U.S. business cycle.

(iv.) All the previously mentioned models are special cases of an *endogenous selection Markov-switching vector autoregressive* model. In an EMS(M, d)-VAR(p) model the transition probabilities $p_{ij}(\cdot)$ are functions of the observed time series vector y_{t-d}:

$$\Pr(s_t = m | s_{t-1} = i, y_{t-d}) = p_{im}(y'_{t-d}\delta).$$

Thus the observed variables contain additional information on the conditional probability distribution of the states:

$$\Pr(s_t | \{s_{t-j}\}_{j=1}^{\infty}) \overset{a.e.}{\neq} \Pr(s_t | \{s_{t-j}\}_{j=1}^{\infty}, \{y_{t-j}\}_{j=1}^{\infty}).$$

Thus the regime generating process is no longer Markovian. In contrast to the SETAR and the STAR model, EMS-VAR models include the possibility that the threshold depends on the last regime, *e.g.* that the threshold for staying in regime 2 is different from the threshold for switching from regime 1 to regime 2 .

The vector autoregressive model with Markov-switching regimes is founded on at least three traditions. The first is the linear time-invariant *vector autoregressive model*, which is the framework for the analysis of the relation of the variables of the system, the dynamic propagation of innovations to the system, and the effects of changes in regime. Secondly, while the basic statistical techniques have been introduced by Baum and Petrie (1966) and Baum, Petrie, Soules and Weiss (1970) for *probabilistic functions of Markov chains*, the MS-VAR model also encompasses older concepts as the *mixture of normal distributions model* attributed to Pearson (1894) and the *hidden Markov-chain model* traced back to Blackwell and Koopmans (1975) and Heller (1965). Thirdly, in econometrics, the first attempt to create Markov-switching regression models were undertaken by Goldfeld and Quandt (1973) which, however, remained rather rudimentary. The first comprehensive approach to the statistical analysis of Markov-switching regression models has been proposed by Lindgren (1978) which is based on the ideas of Baum *et al.* (1970). In time series analysis, the introduction of the Markov-switching model is due to Hamilton (1988), Hamilton (1989) which inspired most recent contributions. Finally, MS-VAR models as a Gaussian vector autoregressive process conditioned on an exogenous regime generating process is closely related to state space models as well as the concept of doubly stochastic processes introduced by Tjøstheim (1986).

7.2 Markov-switching vector autoregressive processes

Markov-switching vector autoregressions can be considered as generalizations of the basic finite order VAR model of order p. Consider the p-th order autoregression for the K-dimensional time series vector $y_t = (y_{1t}, \ldots, y_{Kt})'$, $t = 1, \ldots, T$,

$$y_t = \nu + A_1 y_{t-1} + \ldots + A_p y_{t-p} + u_t, \tag{2}$$

[3] In threshold autoregressive (TAR) processes, the indicator function is defined in a switching variable $z_{t-d},\ d \geq 0$. In addition, indicator variables can be introduced and treated with error-in-variables techniques. Refer for example to Cosslett and Lee (1985) and Kaminsky (1993).

where $u_t \sim \mathrm{IID}(\boldsymbol{0}, \Sigma)$ and $y_0, \ldots, y_{1-p}$ are fixed. Denoting $A(\mathsf{L}) = \boldsymbol{I}_K - A_1\mathsf{L} - \ldots - A_p\mathsf{L}^p$ as the $(K \times K)$ dimensional lag polynomial, we assume that there are no roots on or inside the unit circle $|A(z)| \neq \boldsymbol{0}$ for $|z| \leq 1$ where L is the lag operator, so that $y_{t-j} = \mathsf{L}^j y_t$. If a normal distribution of the error is assumed, $u_t \sim \mathrm{NID}(\boldsymbol{0}, \Sigma)$, equation (2) is known as the intercept form of a stable *Gaussian* VAR(p) model. This can be reparametrized as the mean adjusted form of a VAR model:

$$y_t - \mu = A_1(y_{t-1} - \mu) + \ldots + A_p(y_{t-p} - \mu) + u_t, \tag{3}$$

where $\mu = (\boldsymbol{I}_K - \sum_{j=1}^{p} A_j)^{-1}\nu$ is the $(K \times 1)$ dimensional mean of y_t.

If the time series are subject to shifts in regime, the stable VAR model with its time invariant parameters might be inappropriate. Then, the MS–VAR model might be considered as a general regime-switching framework. The general idea behind this class of models is that the parameters of the underlying data generating process of the *observed* time series vector y_t depend upon the *unobservable* regime variable s_t, which represents the probability of being in a different state of the world.

The description of the data-generating process is not completed by the *observational* equations (6) or (8). A model for the regime generating process has to be formulated which then allows to infer the evolution of regimes from the data. The special characteristic of the Markov-switching model is the assumption that the unobservable realization of the regime $s_t \in \{1, \ldots, M\}$ is governed by a discrete time, discrete state Markov stochastic process, which is defined by the transition probabilities

$$p_{ij} = \Pr(s_{t+1} = j | s_t = i), \quad \sum_{j=1}^{M} p_{ij} = 1 \quad \forall i, j \in \{1, \ldots, M\}. \tag{4}$$

More precisely, it is assumed that s_t follows an irreducible ergodic M state Markov process with the transition matrix

$$\boldsymbol{P} = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1M} \\ p_{21} & p_{22} & \cdots & p_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ p_{11} & p_{12} & \cdots & p_{1M} \end{bmatrix}, \tag{5}$$

where $p_{iM} = 1 - p_{i1} - \ldots - p_{i,M-1}$ for $i = 1, \ldots, M$.

The assumptions of ergodicity and irreducibility are essential for the theoretical properties of MS-VAR models. A comprehensive discussion of the theory of Markov chains with application to Markov-switching models is given by Hamilton (1994b, ch. 22.2). The estimation procedures discussed in Krolzig (1997b, ch.6) and Krolzig (1997b, ch.8) are flexible enough to capture even these degenerated cases, *e.g.* when there is a single jump ("structural break") into the absorbing state that prevails until the end of the observation period.

In generalization of the mean-adjusted VAR(p) model in equation (3) we would like to consider Markov-switching vector autoregressions of order p and M regimes:

$$y_t - \mu(s_t) = A_1(s_t)\,(y_{t-1} - \mu(s_{t-1})) + \ldots + A_p(s_t)\,(y_{t-p} - \mu(s_{t-p})) + u_t, \tag{6}$$

where $u_t \sim \mathrm{NID}(\boldsymbol{0}, \Sigma(s_t))$ and $\mu(s_t), A_1(s_t), \ldots, A_p(s_t), \Sigma(s_t)$ are parameter shift functions describing the dependence of the parameters[4] $\mu, A_1, \ldots, A_p, \Sigma$ on the realized regime s_t, *e.g.*

$$\mu(s_t) = \begin{cases} \mu_1 & \text{if } s_t = 1, \\ \vdots & \\ \mu_M & \text{if } s_t = M. \end{cases} \tag{7}$$

In the model (6) there is after a change in the regime an immediate one–time jump in the process mean. Occasionally, it may be more plausible to assume that the mean smoothly approaches a new level after the transition from one state to another. In such a situation the following model with a regime-dependent intercept term $\nu(s_t)$ may be used:

$$y_t \quad = \quad \nu(s_t) + A_1(s_t)y_{t-1} + \ldots + A_p(s_t)y_{t-p} + u_t. \tag{8}$$

[4] In the notation of state-space models, the varying *parameters* $\mu, \nu, A_1, \ldots, A_p, \Sigma$ become functions of the model's *hyper-parameters*.

Table 1 Markov-Switching Vector Autoregressive Models.

		MSM	MSI Specification		
		μ varying	μ invariant	ν varying	ν invariant
A_j invariant	Σ invariant	MSM–VAR	*linear* MVAR	MSI–VAR	*linear* VAR
	Σ varying	MSMH–VAR	MSH–MVAR	MSIH–VAR	MSH–VAR
A_j varying	Σ invariant	MSMA–VAR	MSA–MVAR	MSIA–VAR	MSA–VAR
	Σ varying	MSMAH–VAR	MSAH–MVAR	MSIAH–VAR	MSAH–VAR

In contrast to the linear VAR model, the mean adjusted form (6) and the intercept form (8) of an MS(M)–VAR(p) model are not equivalent. In Krolzig (1997b, ch.3) it is shown that these forms imply different dynamic adjustments of the observed variables after a change in regime. While a permanent regime shift in the mean $\mu(s_t)$ causes an immediate jump of the observed time series vector onto its new level, the dynamic response to a once-and-for-all regime shift in the intercept term $\nu(s_t)$ is identical to an equivalent shock in the white noise series u_t.

In the most general specification of an MS-VAR model, all parameters of the autoregression are conditioned on the state s_t of the Markov chain such that each regime m VAR(p) parameterisation $\nu(m)$ (or μ_m), Σ_m, $A_{1m}, \ldots, A_{jm}, m = 1, \ldots, M$, such that

$$y_t = \begin{cases} \nu_1 + A_{11} y_{t-1} + \ldots + A_{p1} y_{t-p} + \Sigma_1^{1/2} u_t, & \text{if } s_t = 1 \\ \vdots & \\ \nu_M + A_{1M} y_{t-1} + \ldots + A_{pM} y_{t-p} + \Sigma_M^{1/2} u_t, & \text{if } s_t = M \end{cases}$$

where $u_t \sim \mathrm{NID}(\boldsymbol{0}, \boldsymbol{I}_K)$.

However for empirical applications, it might be more helpful to use a model where only some parameters are conditioned on the state of the Markov chain, while the other parameters are regime invariant. Particular MS-VAR models can be introduced where the autoregressive parameters, the mean or the intercepts, are regime-dependent and where the error term is hetero- or homoskedastic.

The MS-VAR model allows for a great variety of specifications. In order to establish a unique notation for each model, we specify with the general MS(M) term the regime-dependent parameters:

M Markov-switching *mean* ,
I Markov-switching *intercept* term ,
A Markov-switching *autoregressive parameters* ,
H Markov-switching *heteroskedasticity* .

An overview is given in table 1. In many situations MSI(M)–VAR(p) and MSM(M)–VAR(p) models will be sufficient; a regime-dependent covariance structure of the process might be considered as additional feature.[5] To achieve a distinction of VAR models with time-invariant mean and intercept term, we denote the *mean* adjusted form of a vector autoregression as MVAR(p). If exogenous regressors are included into the system, it is denoted MS(M)-VARX(p).

After this introduction of the two components of MS-VAR models, (i.) the Gaussian VAR model as the conditional data generating process and (ii.) the Markov chain as the regime generating process, we briefly sketch the likelihood-based statistical methods.

For a given regime ξ_t and lagged endogenous variables $Y_{t-1} = (y'_{t-1}, y'_{t-2}, \ldots, y'_1, y'_0, \ldots, y'_{1-p})'$ the conditional probability density function of y_t is denoted by $p(y_t|s_t, Y_{t-1})$. It is convenient to assume in (6) and (8) a normal distribution of the error term u_t, so that

$$p(y_t|s_t = \iota_m, Y_{t-1}) \quad = \quad \ln(2\pi)^{-1/2} \ln|\Sigma|^{-1/2} \exp\{(y_t - \bar{y}_{mt})' \Sigma_m^{-1} (y_t - \bar{y}_{mt})\}, \tag{9}$$

[5] Obviously the MSI and the MSM specifications are equivalent if the order of the autoregression is zero. For this so-called hidden Markov-chain model, we prefer the notation MSI(M)-VAR(0) as the MSI(M)-VAR(0) model and MSI(M)-VAR(p) models with $p > 0$ are isomorphic concerning their statistical analysis.

where $\bar{y}_{mt} = \mathsf{E}[y_t|s_t, Y_{t-1}]$ is the conditional expectation of y_t in regime m. Thus the conditional density of y_t for a given regime s_t is normal as in the VAR model defined in equation (2). Thus:

$$y_t\,|s_t = m, Y_{t-1} \quad \sim \quad \mathrm{NID}\left(\bar{y}_{mt}, \Sigma_m\right), \tag{10}$$

where the conditional means $\bar{y}_{mt}$ are summarized in the vector $\bar{y}_t$ which is *e.g.* in MSI specifications of the form

$$\bar{y}_t = \begin{bmatrix} \bar{y}_{1t} \\ \vdots \\ \bar{y}_{Mt} \end{bmatrix} = \begin{bmatrix} \nu_1 + \sum_{j=1}^p A_{1j} y_{t-j} \\ \vdots \\ \nu_M + \sum_{j=1}^p A_{Mj} y_{t-j} \end{bmatrix}.$$

Assuming that the information set available at time $t-1$ consists only of the sample observations and the presample values collected in Y_{t-1} and the states of the Markov chain up to s_{t-1}, the conditional density of y_t is a mixture of normals[6]:

$$\begin{aligned} & p(y_t|s_{t-1} = i, Y_{t-1}) \\ = & \sum_{m=1}^{M} p(y_t|s_{t-1}, Y_{t-1}) \Pr(s_t = m|s_{t-1} = i) \\ = & \sum_{m=1}^{M} \sum_{i=1}^{M} p_{im} \left(\ln(2\pi)^{-\frac{1}{2}} \ln|\Sigma_m|^{-\frac{1}{2}} \exp\{(y_t - \bar{y}_{mt})' \Sigma_m^{-1} (y_t - \bar{y}_{mt})\} \right) \end{aligned} \tag{11}$$

The information about the realization of the Markov chain is collected to the vector ξ_t,

$$\xi_t = \begin{bmatrix} I(s_t = 1) \\ \vdots \\ I(s_t = M) \end{bmatrix},$$

consisting of binary variables where the indicator function $I(s_t = m)$ is defined as:

$$I(s_t = m) = \begin{cases} 1 \text{ if } s_t = m \\ 0 \text{ otherwise,} \end{cases}$$

such that $\mu(s_t) = \sum_{m=1}^M \mu_m I(s_t = m) = \boldsymbol{M}\xi_t$, where $\boldsymbol{M} = [\mu_1, \ldots, \mu_M]$. Thus, ξ_t denotes the unobserved state of the system. Analogously the densities of y_t conditional on s_t and Y_{t-1} can be collected to the vector η_t:

$$\eta_t = \begin{bmatrix} p(y_t|\xi_t = 1, Y_{t-1}) \\ \vdots \\ p(y_t|\xi_t = M, Y_{t-1}) \end{bmatrix}, \tag{12}$$

equation (11) can be written as

$$p(y_t|\xi_{t-1}, Y_{t-1}) = \eta_t' \boldsymbol{P}' \xi_{t-1}. \tag{13}$$

Since the regime is assumed to be unobservable, the relevant information set available at time $t-1$ consists only of the observed time series until time t and the unobserved regime vector ξ_t has to be replaced by the inference $\Pr(\xi_t|Y_\tau)$. These probabilities of being in regime m given an information set Y_τ are denoted $\xi_{mt|\tau}$ and collected in the vector $\hat{\xi}_{t|\tau}$ as

$$\hat{\xi}_{t|\tau} == \begin{bmatrix} \Pr(s_t = 1|Y_\tau) \\ \vdots \\ \Pr(s_t = M|Y_\tau), \end{bmatrix}$$

which allows two different interpretations. First, $\hat{\xi}_{t|\tau}$ denotes the discrete conditional probability distribution of ξ_t given Y_τ. Secondly, $\hat{\xi}_{t|\tau}$ is equivalent to the conditional mean of ξ_t given Y_τ. This is due to the binarity of the elements of ξ_t, which implies that $\mathsf{E}[\xi_{mt}] = \Pr(\xi_{mt} = 1) = \Pr(s_t = m)$.

[6]The reader is referred to Hamilton (1994a) for an excellent introduction into the major concepts of Markov chains and to Titterington, Smith and Makov (1985) for the statistical properties of mixtures of normals.

Thus, the conditional probability density of y_t based upon Y_{t-1} is given by

$$\begin{aligned} p(y_t|Y_{t-1}) &= \int p(y_t, \xi_{t-1}|Y_{t-1}) d\xi_{t-1} \\ &= \int p(y_t|\xi_{t-1}, Y_{t-1}) \Pr(\xi_{t-1}|Y_{t-1}) d\xi_{t-1} \\ &= \eta_t' \boldsymbol{P}' \hat{\xi}_{t-1|t-1}, \end{aligned} \tag{14}$$

where $\int f(x,\xi_t)d\xi_t := \sum_{m=1}^{M} f(x, \xi_t = \iota_m)$ denotes summation over all possible values of ξ_t.

As with the conditional probability density of a single observation y_t in (14) the conditional probability density of the sample can be derived analogously. The techniques of setting-up the likelihood function in practice are introduced in Krolzig (1997b, ch.6). Here we only sketch the basic approach.

For given presample values Y_0, the density of the sample $Y \equiv Y_T$ conditional on the states ξ is determined by

$$p(Y|\xi) = \prod_{t=1}^{T} p(y_t|\xi_t, Y_{t-1}). \tag{15}$$

Hence, the joint probability distribution of observations and states can be calculated as

$$p(Y,\xi) \quad = \quad p(Y|\xi)\ \Pr(\xi) = \prod_{t=1}^{T} p(y_t|\xi_t, Y_{t-1}) \ \prod_{t=2}^{T} \Pr(\xi_t|\xi_{t-1})\ \Pr(\xi_1).$$

Thus, the unconditional density of Y is given by the marginal density

$$p(Y) = \int p(Y,\xi)\ d\xi. \tag{16}$$

The maximization of the likelihood function of an MS-VAR model entails an iterative estimation technique to obtain estimates of the parameters of the autoregression and the transition probabilities governing the Markov chain of the unobserved states. Denote this parameter vector by $\lambda = (\theta, \rho)$, so λ is chosen to maximize the likelihood for given observations $Y_T = (y_T', \ldots, y_{1-p}')'$.

Maximum likelihood (ML) estimation of the model is based on an implementation of the Expectation Maximization (EM) algorithm proposed by Hamilton (1990) for this class of model – an overview on alternative numerical techniques for the maximum likelihood estimation of VAR(M)-MS(p) models is given in Krolzig (1997b). The EM algorithm introduced by Dempster, Laird and Rubin (1977) is designed for a general class of models where the observed time series depends on some unobservable stochastic variables - for MS-AR models these are the regime variable s_t. Each iteration of the EM algorithm consists of two steps. The *expectation* step involves a pass through the filtering and smoothing algorithms, using the estimated parameter vector $\lambda^{(j-1)}$ of the last maximization step in place of the unknown true parameter vector. This delivers an estimate of the smoothed probabilities $\Pr(\xi|Y, \lambda^{(j-1)})$ of the unobserved states ξ_t (where ξ records the history of the Markov chain). In the *maximization* step, an estimate of the parameter vector λ is derived as a solution $\widetilde{\lambda}$ of the first-order conditions associated with the likelihood function, where the conditional regime probabilities $\Pr(\xi|Y,\lambda)$ are replaced with the smoothed probabilities $\Pr(\xi|Y,\lambda^{(j-1)})$ derived in the last expectation step. Equipped with the new parameter vector λ the filtered and smoothed probabilities are updated in the next expectation step, and so on, guaranteeing an increase in the value of the likelihood function at each step.

Regimes constructed in this way are an important instrument for interpreting MS-VAR models. They constitute an optimal inference on the latent state of the economic process, whereby probabilities are assigned to the unobserved regimes conditional on the available information set. It follows by the definition of the conditional density that the conditional distribution of the total regime vector ξ is given by

$$\Pr(\xi|Y) = \frac{p(Y,\xi)}{p(Y)}.$$

Thus, the desired conditional regime probabilities $\Pr(\xi_t|Y)$ can be derived by marginalization of $\Pr(\xi|Y)$. These cumbrous calculations can be simplified by recursive filtering and smoothing algorithms discussed in Krolzig (1997b, ch.5). These statistical tools provide inference for ξ_t given a specified observation set $Y_\tau, \tau \leq T$ which reconstruct the time path of the regime, $\{\xi_t\}_{t=1}^{T}$, under alternative information sets:

$$
\begin{array}{llll}
\hat{\xi}_{t|\tau}, & \tau < t & \textit{predicted} & \text{regime probabilities.} \\
\hat{\xi}_{t|\tau}, & \tau = t & \textit{filtered} & \text{regime probabilities,} \\
\hat{\xi}_{t|\tau}, & t < \tau \leq T & \textit{smoothed} & \text{regime probabilities.}
\end{array}
$$

In practice, mainly the filtered regime probabilities, $\hat{\xi}_{t|t}$, the one-step predicted regime probabilities $\hat{\xi}_{t|t-1}$, and the full-sample smoothed regime probabilities, $\hat{\xi}_{t|T}$, are considered.

The MS-VAR model provides a very flexible framework which allows for heteroskedasticity, occasional shifts, reversing trends, and forecasts performed in a non-linear manner. The implications of particular MS-VAR models for their estimation are discussed in Krolzig (1997b, ch.9).

8 Model formulation

Model formulation is based on the names of variables. The following steps are involved in model formulation:

- Create a MSVAR object.
- Load your data into the MSVAR database using the facilities of the Database class.
- Transform the data.
- Use `Select` to formulate the model. A constant will be included by default.
- Use `SetSample` to specify the sample.
- Use `SetModel` for a different specification of the model, for example:
  - time-invariant intercept;
  - regime-dependent intercept;
  - regime-dependent mean.
- For reports of the progress of the EM algorithm, use `SetPrint`.
- For changes of convergence threshold and the maximum number of iteration, use `SetEmOptions`.
- By default, a graphic presentation of the results is shown, standard errors are calculated, and gwg files of all givewin graphics saved. Use `SetOptions` to change this.
- Finally, use `Estimate` for estimation.

9 Examples

9.1 Hamilton's model of the US business cycle

MSVAR can be used to compute ML estimates of univariate and multivariate MS-VAR models. The first example replicates Hamilton (1989) and is provided in the file `hamilton.ox`.

```
/---------------------------------- hamilton.ox -----------------------------------/
#include <oxstd.h>
#import<maximize>
#import<database>
#import<hmk>
#import<msvar>

main()
{
    decl msvar = new MSVAR();
    msvar->LoadIn7("gnp82.in7");
    msvar->Select(Y_VAR, { "DUSGNP", 0, 4});
    msvar->SetSample(1900,1,1999,4);
    msvar->SetModel(MSM, 2);
    msvar->Estimate();
}
/---------------------------------- hamilton.ox -----------------------------------/
```

The Hamilton (1989) model of the US business cycle fostered a great deal of interest in the MS–AR model as an empirical vehicle for characterizing macroeconomic fluctuations, and there have been a number of subsequent

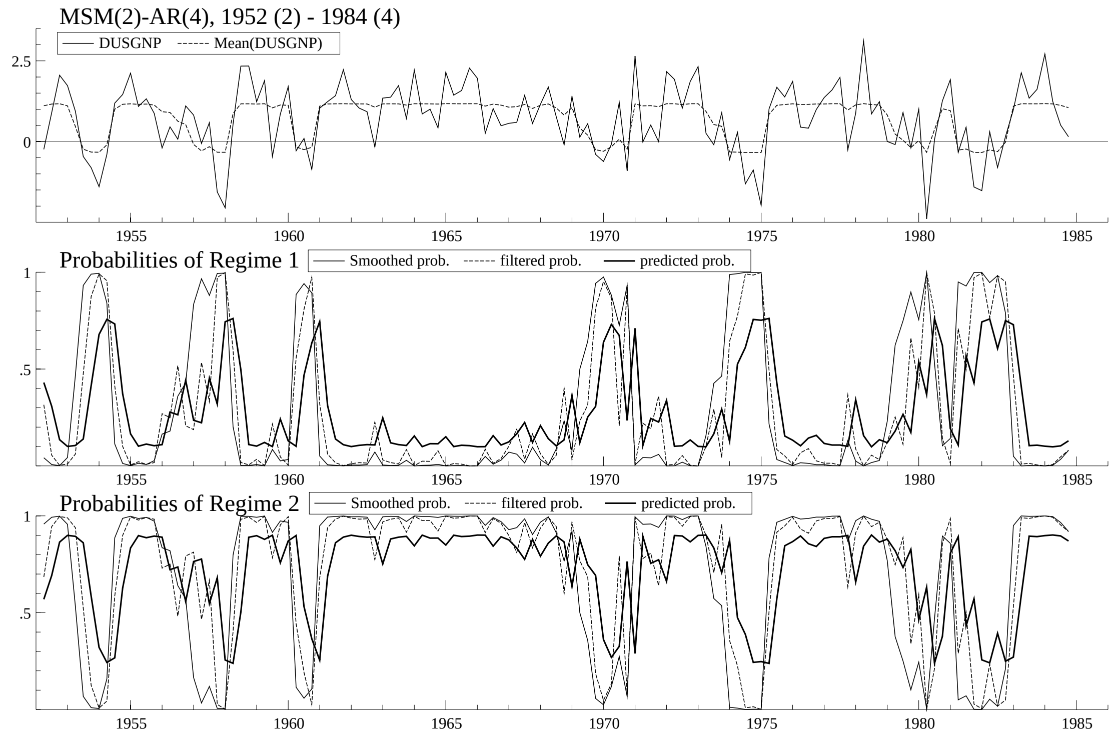

Figure 1 Hamilton model.

extensions and refinements (see the literature discussed in Krolzig, 1997b). The Hamilton (1989) model of the US business cycle is an MSM(2)-AR(4) of the quarterly percentage change in US real GNP from 1953 to 1984:

$$\Delta y_t - \mu(s_t) = \alpha_1 \left(\Delta y_{t-1} - \mu(s_{t-1})\right) + \ldots + \alpha_4 \left(\Delta y_{t-p} - \mu(s_{t-4})\right) + u_t, \tag{17}$$

where $u_t \sim \mathsf{NID}(0, \sigma^2)$, and the conditional mean $\mu(s_t)$ switches between two states:

$$\mu(s_t) = \begin{cases} \mu_1 < 0 & \text{if } s_t = 1 \text{ (‘expansion’ or ‘boom’)}, \\ \mu_2 > 0 & \text{if } s_t = 2 \text{ (‘contraction’ or ‘recession’)}. \end{cases}$$

The variance of the disturbance term, σ^2, is assumed to be the same in both regimes. Thus, contractions and expansions are modelled as switching regimes of the stochastic process generating the growth rate of real GNP. The regimes are associated with different conditional distributions of the growth rate of real GNP, where, for example, the mean is positive in the first regime (‘expansion’) and negative in the second regime (‘contraction’). The transition probabilities are constant:

$$\begin{aligned} p_{21} &= \Pr(\text{ contraction in } t \mid \text{expansion in } t-1), \\ p_{12} &= \Pr(\text{ expansion in } t \mid \text{contraction in } t-1). \end{aligned}$$

For a given parametric specification of the model, probabilities are assigned to the unobserved regimes ‘expansion’ and ‘contraction’ conditional on the available information set which constitute an optimal inference on the latent state of the economy. Regimes reconstructed in this way are an important instrument for interpreting business cycles using MS-(V)AR models. The graphical representation of the filtered, $Pr(s_t|Y_t)$, smoothed, $Pr(s_t|Y_T)$ and predicted regime probabilities, $Pr(s_t|Y_{t-1})$, as in figure 1 is automatically produced (but can be switched off by `SetOptions()`).

The output produced by MSVAR is given by the following:

```
/------------------------------- hamilton.out -------------------------------/

MSVAR0.98 (c) Hans-Martin Krolzig, 06-12-1998
Object created 11-12-1998,  4:33:34

****    EM algorithm converged after 43 iterations  ****

Estimation Results for the MSM(2)-AR(4) Model, 1952 (2) - 1984 (4)

no. obs. per eq. :        131    in the system :        131
no. parameters   :          9    linear system :          6
no. restrictions :          1
no. nuisance p.  :          2

log-likelihood   :  -181.4236    linear system :  -183.6692

AIC criterion    :     2.9072    linear system :     2.8957
HQ  criterion    :     2.9875    linear system :     2.9492
SC  criterion    :     3.1048    linear system :     3.0274

LR linearity test:     4.4911    Chi(1) =[0.0341] *   Chi(3) =[0.2131]  DAVIES=[0.2131]

****     transition matrix  ****

                  Regime 1     Regime 2
Regime 1           0.76202     0.098594
Regime 2           0.23798      0.90141

Note that p[i][j]=Pr{s(t)=i|s(t-1)=j}

                      nObs        Prob.     Duration
Regime 1            38.451      0.29293       4.2020
Regime 2            92.549      0.70707       10.143


****     coefficients   ****
                    DUSGNP
Mean(Reg.1)       -0.34027
Mean(Reg.2)         1.1727
DUSGNP_1          0.010772
DUSGNP_2         -0.062674
DUSGNP_3          -0.24615
DUSGNP_4          -0.20087

****     variance   ****
                    DUSGNP
Variance           0.59270

****     dynamics   ****

                  DUSGNP_1     DUSGNP_2     DUSGNP_3     DUSGNP_4
DUSGNP            0.010772    -0.062674     -0.24615     -0.20087

Eigenvalues of the companion matrix

         real      complex      modulus
      0.46991      0.61766      0.77609
      0.46991     -0.61766      0.77609
     -0.46452      0.34309      0.57749
     -0.46452     -0.34309      0.57749
Cannot show draw window!

****     Calculate numerical second-order derivatives   ****

****     Calculate covariance matrix    ****

****     standard errors    *****

                    DUSGNP
Mean(Reg.1)        0.24409
Mean(Reg.2)        0.14233
DUSGNP_1          0.089525
DUSGNP_2          0.081071
DUSGNP_3          0.085861
DUSGNP_4          0.086692

****     t - values     *****

                    DUSGNP
Mean(Reg.1)         1.3940
Mean(Reg.2)         8.2395
DUSGNP_1           0.12033
DUSGNP_2           0.77308
DUSGNP_3            2.8669
DUSGNP_4            2.3170

/------------------------------- hamilton.out -------------------------------/
```

9.2 An MS-VAR model of international business cycles

This example demonstrates the easy use of MSVAR for modelling multiple time series. It reconsiders the MS-VAR model of the common business cycle of six OECD countries of four contintents proposed by Krolzig (1997a) and is provided in the file `wbc.ox`.

```
/-------------------------------- wbc.ox ----------------------------------/
#include <oxstd.h>
#import<maximize>
#import<database>
#import<hmk>
#import<msvar>

main()
{
    format(120);

    decl msvar = new MSVAR();
    msvar->LoadIn7("wbc.in7");
    decl p=1;
    msvar->Select(Y_VAR, {  "DYUSA", 0, p, "DYCAN", 0, p, "DYAUS", 0, p,
                            "DYUK",  0, p, "DYFRG", 0 ,p, "DYJAP", 0, p  });
    msvar->SetSample(1962, 1, 1991, 4);
    msvar->SetModel(MSMH, 3)
    msvar->Estimate();
}
/-------------------------------- wbc.ox ----------------------------------/
```

The model is an MSMH(3)-VAR(1) of the six-dimensional vector Δx_t of real GNP (GDP) growth rates:

$$\Delta x_t - \mu(s_t) = A_1 \left(\Delta x_{t-1} - \mu(s_{t-1})\right) + u_t. \quad (18)$$

where $u_t|s_t \sim \mathsf{NID}(0, \Sigma)s_t))$.

The output produced by MSVAR:

```
/-------------------------------- wbc.out ----------------------------------/

MSVAR0.98 (c) Hans-Martin Krolzig, 06-12-1998
Object created 11-12-1998,  6:38:33

**** EM algorithm converged after 15 iterations ****

Estimation Results for the MSMH(3)-VAR(1) Model, 1962 (1) - 1991 (4)

no. obs. per eq. :        120    in the system :        720
no. parameters   :        123    linear system :         63
no. restrictions :         54
no. nuisance p.  :          6

log-likelihood   :  -944.5186    linear system : -1031.6696

AIC criterion    :    17.7920    linear system :    18.2445
HQ  criterion    :    18.9523    linear system :    18.8388
SC  criterion    :    20.6492    linear system :    19.7079

LR linearity test:   174.3020    Chi(54) =[0.0000] **  Chi(60)=[0.0000] **  DAVIES=[0.0000] **

****  transition matrix  ****

                  Regime 1     Regime 2     Regime 3
Regime 1           0.84001     0.035318     0.098981
Regime 2          0.065903      0.91106      0.10045
Regime 3          0.094090     0.053621      0.80057

Note that p[i][j]=Pr{s(t)=i|s(t-1)=j}

                      nObs        Prob.     Duration
Regime 1            31.768      0.26364       6.2503
Regime 2            57.715      0.48230       11.244
Regime 3            30.517      0.25406       5.0142

****  coefficients  ****
                     DYUSA        DYCAN        DYAUS         DYUK        DYFRG        DYJAP
Mean(Reg.1)       -0.20451    -0.017831     0.095027     -0.24936      0.14609      0.74188
Mean(Reg.2)        0.98725       1.1143      0.93683      0.70030      0.83433       1.1601
Mean(Reg.3)         1.1572       1.8868       1.8791       1.4220       1.3405       2.7402
DYUSA_1            0.12652      0.37727     0.083398    -0.063324      0.28768     0.079948
DYCAN_1            0.14197     0.084284      0.10809      0.11918    -0.034131     -0.15884
DYAUS_1            0.13342      0.26827    -0.076156     -0.10132    -0.023398     0.091573
```

```
DYUK_1            0.20504      0.26237     0.053264     -0.29617     0.029327     0.040775
DYFRG_1         -0.045128      0.12760     0.071787    -0.091827     -0.13934      0.10923
DYJAP_1          0.017786     -0.11795     0.022078     -0.27323   -0.0012243     -0.22113

****  variance  ****
Regime 1: variance (det =0.0406462)
              DYUSA        DYCAN        DYAUS         DYUK        DYFRG        DYJAP
DYUSA       0.91587      0.23235     0.083618     0.060401      0.30217      0.50288
DYCAN       0.23235      0.65867      0.35453     -0.15552    -0.069157     -0.14047
DYAUS      0.083618      0.35453       1.0113     -0.11064     -0.15673     -0.11238
DYUK       0.060401     -0.15552     -0.11064      0.71303     -0.10513      0.23752
DYFRG       0.30217    -0.069157     -0.15673     -0.10513      0.65723      0.34879
DYJAP       0.50288     -0.14047     -0.11238      0.23752      0.34879      0.76293

Correlation
              DYUSA        DYCAN        DYAUS         DYUK        DYFRG        DYJAP
DYUSA        1.0000      0.29916     0.086885     0.074744      0.38948      0.60159
DYCAN       0.29916       1.0000      0.43438     -0.22694     -0.10511     -0.19816
DYAUS      0.086885      0.43438       1.0000     -0.13029     -0.19224     -0.12794
DYUK       0.074744     -0.22694     -0.13029       1.0000     -0.15357      0.32203
DYFRG       0.38948     -0.10511     -0.19224     -0.15357       1.0000      0.49256
DYJAP       0.60159     -0.19816     -0.12794      0.32203      0.49256       1.0000

Regime 2: variance (det =0.0834924)
              DYUSA        DYCAN        DYAUS         DYUK        DYFRG        DYJAP
DYUSA       0.57665      0.20971     0.019416     -0.14887     -0.16995    -0.046502
DYCAN       0.20971      0.44209     -0.16904    -0.017633    -0.064040     0.032231
DYAUS      0.019416     -0.16904       1.0763     -0.15005     -0.14683     0.011486
DYUK       -0.14887    -0.017633     -0.15005       1.2738      0.48963    0.0015602
DYFRG      -0.16995    -0.064040     -0.14683      0.48963       1.0262     0.013485
DYJAP     -0.046502     0.032231     0.011486    0.0015602     0.013485      0.42004

Correlation
              DYUSA        DYCAN        DYAUS         DYUK        DYFRG        DYJAP
DYUSA        1.0000      0.41534     0.024645     -0.17369     -0.22093    -0.094487
DYCAN       0.41534       1.0000     -0.24505    -0.023497    -0.095076     0.074795
DYAUS      0.024645     -0.24505       1.0000     -0.12814     -0.13971     0.017082
DYUK       -0.17369    -0.023497     -0.12814       1.0000      0.42824    0.0021330
DYFRG      -0.22093    -0.095076     -0.13971      0.42824       1.0000     0.020539
DYJAP     -0.094487     0.074795     0.017082    0.0021330     0.020539       1.0000

Regime 3: variance (det =0.522656)
              DYUSA        DYCAN        DYAUS         DYUK        DYFRG        DYJAP
DYUSA       0.34930    -0.028082     -0.14505      0.32832    -0.099769     -0.15917
DYCAN     -0.028082      0.46940     0.042784      0.45118      0.19908    -0.060313
DYAUS      -0.14505     0.042784       1.0358      0.48427      0.28199     -0.18167
DYUK        0.32832      0.45118      0.48427       2.4906      0.40783    -0.018471
DYFRG     -0.099769      0.19908      0.28199      0.40783       3.7553      0.61910
DYJAP      -0.15917    -0.060313     -0.18167    -0.018471      0.61910       1.0320

Correlation
              DYUSA        DYCAN        DYAUS         DYUK        DYFRG        DYJAP
DYUSA        1.0000    -0.069353     -0.24115      0.35201    -0.087110     -0.26511
DYCAN     -0.069353       1.0000     0.061358      0.41728      0.14994    -0.086656
DYAUS      -0.24115     0.061358       1.0000      0.30151      0.14298     -0.17572
DYUK        0.35201      0.41728      0.30151       1.0000      0.13335    -0.011521
DYFRG     -0.087110      0.14994      0.14298      0.13335       1.0000      0.31448
DYJAP      -0.26511    -0.086656     -0.17572    -0.011521      0.31448       1.0000

****   dynamics  ****

VAR matrix at lag 1
              DYUSA        DYCAN        DYAUS         DYUK        DYFRG        DYJAP
DYUSA       0.12652      0.14197      0.13342      0.20504    -0.045128     0.017786
DYCAN       0.37727     0.084284      0.26827      0.26237      0.12760     -0.11795
DYAUS      0.083398      0.10809    -0.076156     0.053264     0.071787     0.022078
DYUK      -0.063324      0.11918     -0.10132     -0.29617    -0.091827     -0.27323
DYFRG       0.28768    -0.034131    -0.023398     0.029327     -0.13934   -0.0012243
DYJAP      0.079948     -0.15884     0.091573     0.040775      0.10923     -0.22113

Eigenvalues of the companion matrix

         real      complex      modulus
      0.44944      0.00000      0.44944
     -0.32818     -0.11072      0.34636
     -0.32818      0.11072      0.34636
    -0.091935     -0.15895      0.18363
    -0.091935      0.15895      0.18363
     -0.13121      0.00000      0.13121

****    Automatic calculation of standard errors has been suspended as
        the model has 123 parameters. Use StdErr()                      ****

/---------------------------------- wbc.out ----------------------------------/
```

9.3 A Markov-switching vector equilibrium correction model

MSVAR can be used for the maximum likelihood estimation of Markov-switching vector equilibrium correction models (MS-VECM) proposed by Krolzig (1996). This example replicates the results of the MS-VECM of US Output and Employment considered by Krolzig and Toro (1998a), and is provided in the file `kroto.ox`.

An MS-VECM is a vector equilibrium correction model with shifts in the drift $\delta(s_t)$ and in the long-run equilibrium $\mu(s_t)$:

$$\Delta x_t - \delta(s_t) = \alpha\left(\beta' x_{t-1} - \mu(s_t) - \gamma t\right) + \sum_{k=1}^{p-1} A_i\left(\Delta x_{t-k} - \delta(s_t)\right) + u_t \tag{19}$$

and the innovations u_t are conditionally Gaussian, $u_t|s_t \sim \mathsf{NID}(\boldsymbol{0}, \Sigma(s_t))$. The parameters δ and μ depend upon a stochastic, unobservable regime variable $s_t \in \{1, \ldots, M\}$. As in the previous examples, the stochastic process for generating the unobservable regimes is an ergodic Markov chain defined by the transition probabilities

$$p_{ij} = \Pr(s_{t+1} = j | s_t = i), \quad \sum_{j=1}^{M} p_{ij} = 1 \quad \forall i, j \in \{1, \ldots, M\}. \tag{20}$$

By inferring the probabilities of the unobserved regimes conditional on an available information set, it is then possible to reconstruct the regimes.

By following the two-stage procedure proposed in Krolzig (1996), the cointegration properties of the output, y_t, and employment, n_t, data are studied within a linear vector autoregressive representation using maximum likelihood techniques (as provided by the `PcFiml` class). Conditional on the estimated cointegration matrix, we get the following representation:

$$\Delta x_t = \nu(s_t) + A_1 x_{t-1} + \alpha z_{t-1} + u_t. \tag{21}$$

The stationary linear transform $z_{t-1} = y_{t-1} - n_{t-1} - \tilde{\gamma} t - \bar{\mu}$ has been normalized such that $\mathsf{E}[z_t] = 0$, stored in the database and introduced to the system as an exogenous variable by using `Select(X_VAR, { "zName", 1, 1})`.

Maximum likelihood (ML) estimation of the MSIH-VARX model (21) is then based on the MSVAR class using a version of the Expectation-Maximization (EM) algorithm discussed in Hamilton (1990) and Krolzig (1997b).

```
/--------------------------------- kroto.ox ----------------------------------/
#include <oxstd.h>
#import<maximize>
#import<database>
#import<hmk>
#import <msvar>

main()
{
    decl time=timer();
    decl msvar = new MSVAR();

    msvar->LoadIn7("kroto.in7");
    msvar->SetOptions(FALSE,FALSE,TRUE);        // settings
            //(automatic StdErrors, DrawResults, save gwg files)

    msvar->SetPrint(TRUE,TRUE);                 // all results are printed
    msvar->SetEstimate(1e-5, 100, 1);           // EmAlg specification
            //(tolerance, max.#iterations, max.#iterations for MSteps)

    decl M=3;                                   // number of regimes
    decl p=1;                                   // number of lages

    msvar->Select(Y_VAR, { "DN", 0, p, "DY", 0, p});
    msvar->Select(X_VAR, { "Cyn", 1, 1});
    msvar->SetSample(1962,1,1997,4);
    msvar->Model(MSIH, M);         // model specification (Model,#regimes)

            // Model={MSI,MSIH,MSM,MSMH,MSIA,MSIAH,MSIA,MSIAH,MSH}
            // allowing for shifts in the (I)ntercepts, (M)ean, (A)utoregressive
            // parameters and (H)eteroskedasticity
```

```
    msvar->Estimate();                          // estimates
    msvar->DrawResults();                       // shows graphics
    msvar->DrawErrors();                        // shows graphics
    msvar->DrawFit();                           // shows graphics
    msvar->StdErr();                            // calculates standard errors

    delete msvar;
    print("\n\n****\ttime passed: ",    timespan(time), "\t****\n");
```

/-------------------------------- kroto.ox ----------------------------------/

`kroto.ox` shows how the estimation can be implemented in MSVAR. The figures and tables produced by the program follow.

/-------------------------------- kroto.out ----------------------------------/

```
MSVAR.OX (c) Hans-Martin Krolzig, 06-08-1998
Object created  4-12-1998,  1:57:04

****   Calculate starting values   ****

It.  0  LogLik =  -129.2101  Pct.Change =100.0000
It.  1  LogLik =  -119.1349  Pct.Change =  7.7975
It.  2  LogLik =  -115.3296  Pct.Change =  3.1941
It.  3  LogLik =  -114.1886  Pct.Change =  0.9894
It.  4  LogLik =  -113.9093  Pct.Change =  0.2446
It.  5  LogLik =  -113.8320  Pct.Change =  0.0679

.................................................

It. 39  LogLik =  -112.2400  Pct.Change =  0.0032
It. 40  LogLik =  -112.2385  Pct.Change =  0.0013
It. 41  LogLik =  -112.2379  Pct.Change =  0.0005

****    EM algorithm converged after 42 iterations ****


Estimation Results for the MSIH(3)-VARX(1) Model, 1962 (3) - 1997 (1)

no. obs. per eq. :        139    in the system :        278
no. parameters   :         27    linear system :         11
no. restrictions :         10
no. nuisance p.  :          6

log-likelihood   :  -112.2379    linear system :  -145.4374

AIC criterion    :     2.0034    linear system :     2.2509
HQ  criterion    :     2.2351    linear system :     2.3453
SC  criterion    :     2.5734    linear system :     2.4831

LR linearity test:    66.3990    Chi(10) =[0.0000] **
                                 Chi(16) =[0.0000] **   DAVIES  =[0.0000] **

****    transition matrix  ****

                  Regime 1     Regime 2     Regime 3
Regime 1           0.83038     0.051274     0.030876
Regime 2          0.035658      0.94834     0.068966
Regime 3           0.13396   0.00038374      0.90016

Note that p[i][j]=Pr{s(t)=i|s(t-1)=j}

                      nObs        Prob.     Duration
Regime 1            28.561      0.20637       5.8955
Regime 2            64.463      0.51476       19.358
Regime 3            45.975      0.27887       10.016


****    coefficients  ****
                        DN           DY
Const(Reg.1)     -0.037737     -0.22234
Const(Reg.2)       0.23522      0.61034
Const(Reg.3)       0.43127       1.1774
DN_1               0.53512      0.13230
DY_1              0.022430   0.00027807
Cyn_1             0.073950    -0.013330

****    variance  ****
Regime 1: variance (det =0.0623182)
                 DN           DY
DN          0.20525      0.40077
DY          0.40077       1.0862

Correlation
                 DN           DY
DN           1.0000      0.84880
DY          0.84880       1.0000
```

```
Regime 2: variance (det =0.00289041)
                 DN            DY
DN         0.018729      0.022949
DY         0.022949       0.18245

Correlation
                 DN            DY
DN           1.0000       0.39259
DY          0.39259        1.0000

Regime 3: variance (det =0.0285466)
                 DN            DY
DN         0.083341       0.15188
DY          0.15188       0.61930

Correlation
                 DN            DY
DN           1.0000       0.66851
DY          0.66851        1.0000

****    dynamics  ****

VAR matrix at lag 1
                 DN            DY
DN          0.53512      0.022430
DY          0.13230    0.00027807

Eigenvalues of the companion matrix

         real
      0.54061
   -0.0052139

****    Calculate numerical second-order derivatives  ****

****    Calculate covariance matrix  ****

****    standard errors  *****

                        DN            DY
Const(Reg.1)       0.11021       0.27604
Const(Reg.2)      0.042897       0.11384
Const(Reg.3)      0.084215       0.21030
DN_1              0.059509       0.16819
DY_1              0.036103      0.097148
Cyn_1             0.022969      0.067082

****    t - values  *****

                        DN            DY
Const(Reg.1)       0.34241       0.80549
Const(Reg.2)        5.4834        5.3616
Const(Reg.3)        5.1211        5.5987
DN_1                8.9921       0.78662
DY_1               0.62129     0.0028624
Cyn_1               3.2195       0.19871

/--------------------------------- kroto.out -----------------------------------/
```

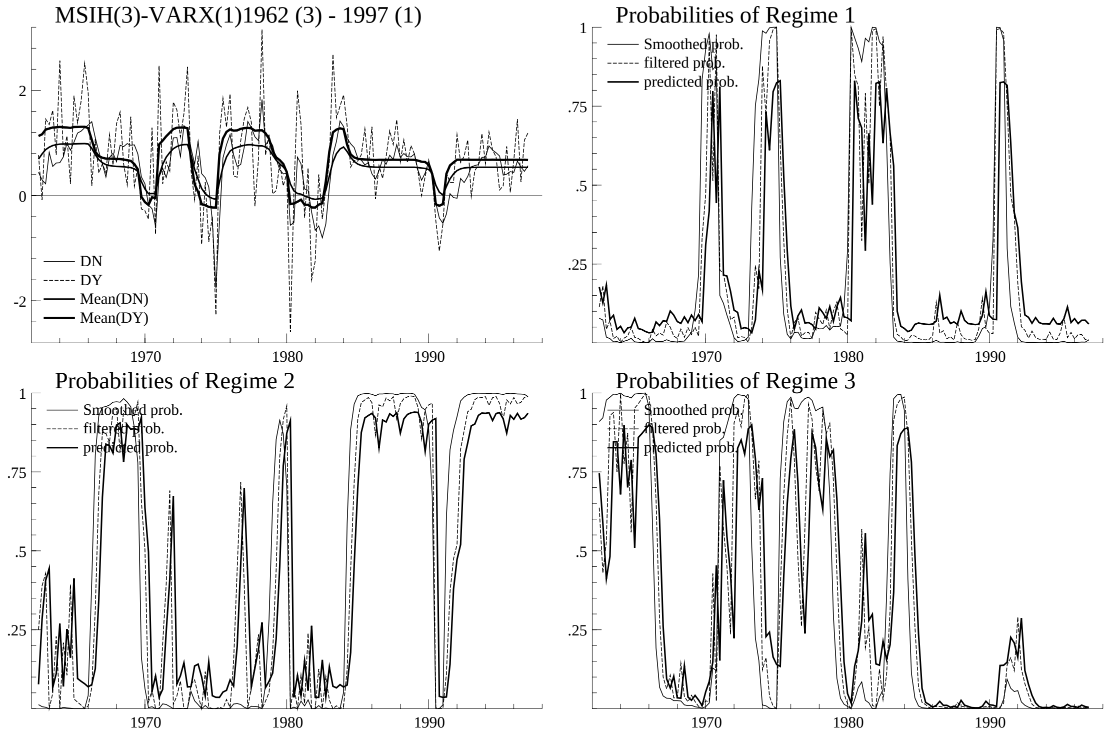

Figure 2 MSVAR Graphics by DrawResults. Regime probabilities.

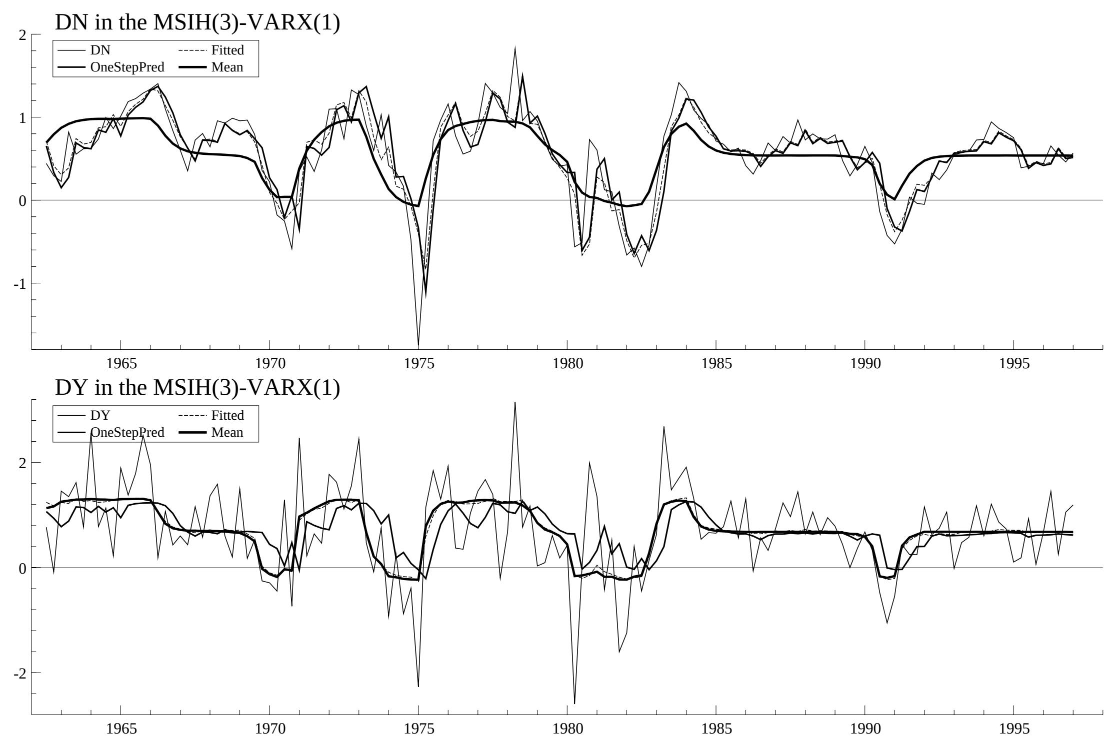

Figure 3 MSVAR Graphics by DrawFit. Actual and fitted values.

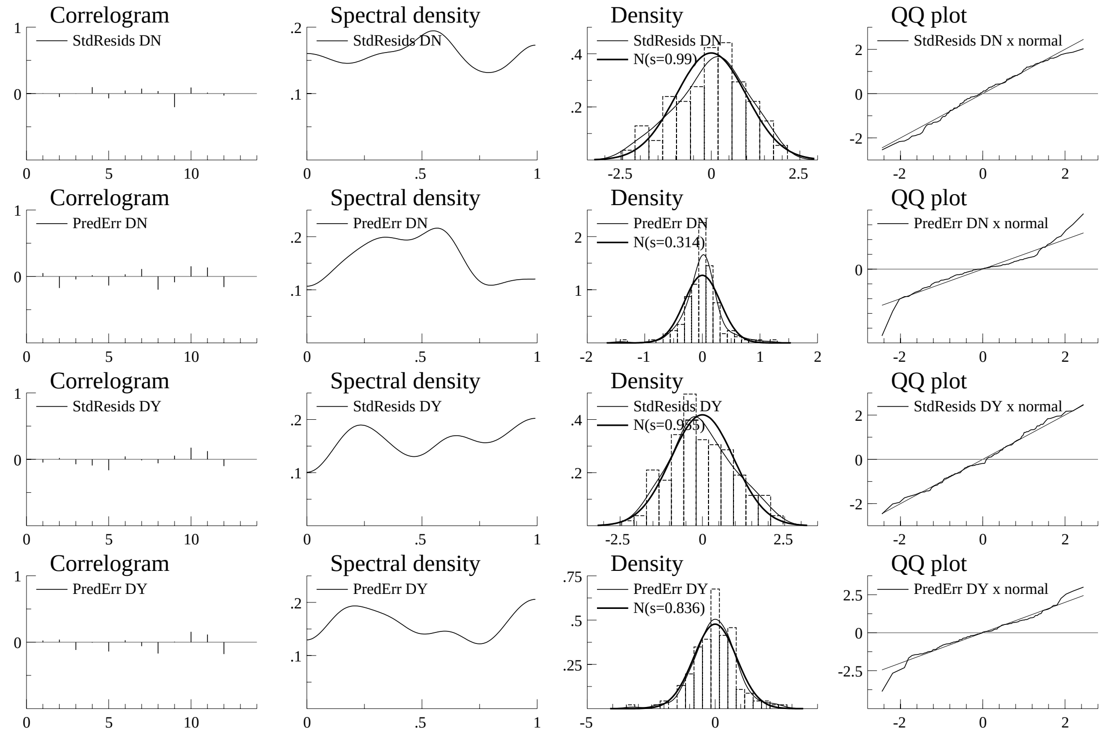

Figure 4 MSVAR Graphics by DrawErrors. Residuals and 1-step prediction errors.

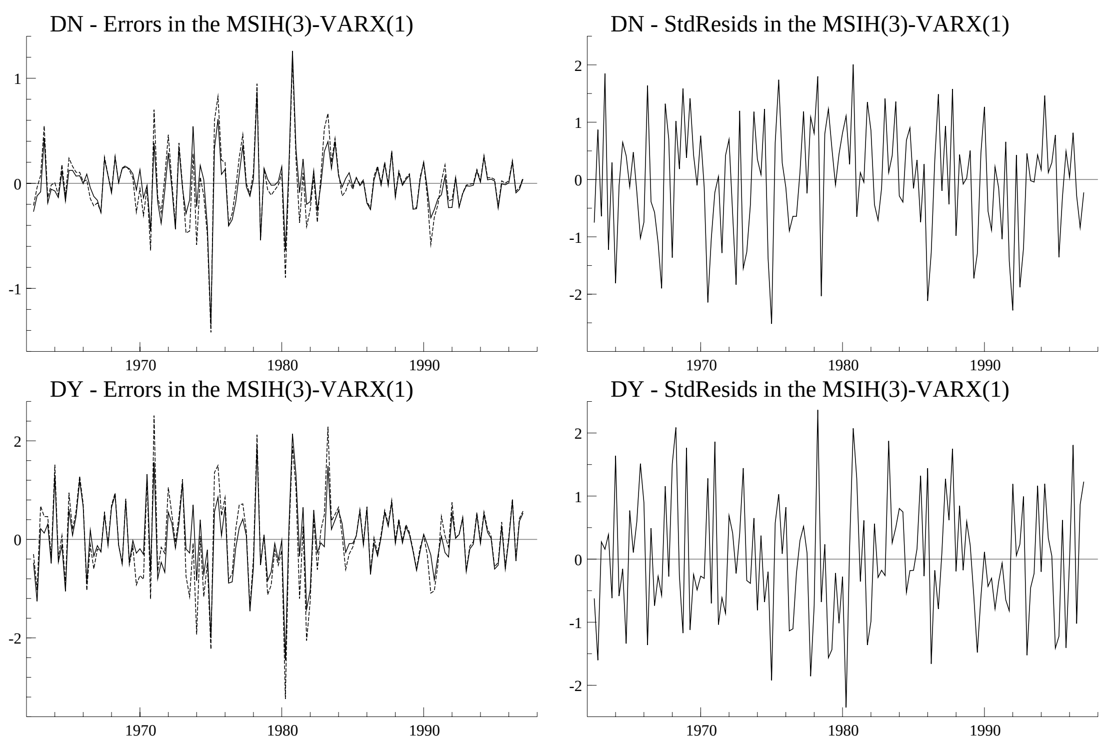

Figure 5 MSVAR Graphics by DrawErrors. Error analysis.

10 Notes and remarks

To do

- Forecasting, see Krolzig (1998) and Clements and Krolzig (1998b)
- Impulse-response analysis, see: Krolzig and Toro (1998a)
- Cointegration analysis, see Krolzig (1996)
- Mu-Delta-Decomposition, see Krolzig and Sensier (1998)
- ARMA representation based model specification, see Krolzig (1997b, ch.3)
- Multi-move Gibbs sampler, see Krolzig (1997b, ch.8)
- Tests for cobreaking, see Krolzig and Toro (1998b)
- Encompassing tests
- Asymmetry tests, see: Clements and Krolzig (1998a)

References

Baum, L. E., and Petrie, T. (1966). Statistical inference for probabilistic functions of finite state Markov chains. *Annals of Mathematical Statistics*, **37**, 1554–1563.

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*, **41**, 164–171.

Blackwell, E., and Koopmans, L. (1975). On the identifiability problem for functions of finite Markov chains. *Annals of Mathematical Statistics*, **28**, 1011–1015.

Clements, M. P., and Krolzig, H. M. (1998a). Business cycle asymmetries: Characterisation and testing based on Markov-switching autoregressions. Mimeo, Institute of Economics and Statistics, University of Oxford.

Clements, M. P., and Krolzig, H. M. (1998b). A comparison of the forecast performance of Markov-switching and threshold autoregressive models of US GNP. *Econometrics Journal*, **1**, C47–C75.

Cosslett, S. R., and Lee, L.-F. (1985). Serial correlation in latent discrete variable models. *Journal of Econometrics*, **27**, 79–97.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood estimation from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, **39, Series B**, 1–38.

Doornik, J. A. (1998). *Object-Oriented Matrix Programming using Ox 2.0*. London: Timberlake Consultants Press.

Doornik, J. A., and Hendry, D. F. (1997). *Modelling Dynamic Systems using PcFiml 9 for Windows*. London: Timberlake Consultants Press.

Goldfeld, S. M., and Quandt, R. E. (1973). A Markov model for switching regressions. *Journal of Econometrics*, **1**, 3–16.

Granger, C. W. J., and Teräsvirta, T. (1993). *Modelling nonlinear economic relationships*: Oxford University Press.

Hamilton, J. D. (1988). Rational-expectations econometric analysis of changes in regime. An investigation of the term structure of interest rates. *Journal of Economic Dynamics and Control*, **12**, 385–423.

Hamilton, J. D. (1989). A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica*, **57**, 357–384.

Hamilton, J. D. (1990). Analysis of time series subject to changes in regime. *Journal of Econometrics*, **45**, 39–70.

Hamilton, J. D. (1994a). State-space models. In Engle, R., and McFadden, D. (eds.), *Handbook of Econometrics*, Vol. 4. Amsterdam: North–Holland.

Hamilton, J. D. (1994b). *Time Series Analysis*. Princeton: Princeton University Press.

Heller, A. (1965). On stochastic processes derived from Markov chains. *Annals of Mathematical Statistics*, **36**, 1286–1291.

Kaminsky, G. (1993). Is there a peso problem? Evidence from the dollar/pound exchange rate, 1976-1987. *American Economic Review*, **83**, 450–472.

Krolzig, H.-M. (1996). Statistical analysis of cointegrated VAR processes with Markovian regime shifts. SFB 373 Discussion Paper 25/1996, Humboldt Universität zu Berlin.

Krolzig, H.-M. (1997a). International business cycles: Regime shifts in the stochastic process of economic growth. Applied Economics Discussion Paper 194, University of Oxford.

Krolzig, H.-M. (1997b). *Markov Switching Vector Autoregressions. Modelling, Statistical Inference and Application to Business Cycle Analysis*. Berlin: Springer.

Krolzig, H.-M. (1998). Predicting Markov-switching vector autoregressive processes. Mimeo, Institute of Economics and Statistics, University of Oxford.

Krolzig, H.-M., and Sensier, M. (1998). A disaggregated markov-switching model of the business cycle in uk manufacturing. Discussion paper 9812, School of Economic Studies, University of Manchester.

Krolzig, H.-M., and Toro, J. (1998a). A new approach to the analysis of shocks and the cycle in a model of output and employment. Discussion Paper, Institute of Economics and Statistics, University of Oxford.

Krolzig, H.-M., and Toro, J. (1998b). Testing for cobreaking and superexogeneity in the presence of deterministic shifts. Discussion Paper, Institute of Economics and Statistics, University of Oxford.

Lindgren, G. (1978). Markov regime models for mixed distributions and switching regressions. *Scandinavian Journal of Statistics*, **5**, 81–91.

Pearson, K. (1894). Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society*, **185**, 71–110.

Sims, C. A. (1980). Macroeconomics and reality. *Econometrica*, **48**, 1–48.

Teräsvirta, T., and Anderson, H. (1992). Modelling nonlinearities in business cycles using smooth transition autoregressive models. *Journal of Applied Econometrics*, **7**, S119–S136.

Titterington, D. M., Smith, A. F. M., and Makov, U. E. (1985). *Statistical Analysis of Finite Mixture Distributions*: New York: Wiley.

Tjøstheim, D. (1986). Some doubly stochastic time series models. *Journal of Time Series Analysis*, **7**, 51–72.

A Glossary of MSVAR functions

The documentation only includes the exported member functions of MSVAR. The non-exported member functions are not documented here as they are only called from other MSVAR function members. Some functions are quite complex, and should be approached with care.

Notation:

K	number of endogenous variables,
M	number of regimes,
N	dimension of the state vector (MSIxx: M, MSMxx: M^{p+1})
p	order of the VAR,
R	number of regressors (excluding constant),
T	number of observations.

MSVAR::DeSelect

```
DeSelect();
```

No return value.

Description

Clears the model formulation, i.e. clears previous calls to `Select()` and `SetSample()`.

MSVAR::Estimate

```
Estimate(const mMu, const mB, const m_mSigma, const Trans);
Estimate(const mProbSt);
Estimate(const );
```

`mMu`	in:	$K \times M$ matrix of means or intercepts (MSAx: $K \times 1$)
`mB`	in:	$K \times R$ matrix of coefficients (MSxAx: $K \times MR$)
`mSigma`	in:	$K \times M$ variance matrix (MSxxH: $K \times MK$)
`mTrans`	in:	$M \times M$ transition matrix (transposed matrix of transition probabilities)
`mProbSt`	in:	$M \times T$ matrix of initial regime probabilities

No return value.

Description

Estimates the model and prints the results, unless this is switched off by `SetPrint()`. Use `Select`, `SetSample` and `SetModel` prior to `Estimate` to formulate the model.
When initial parameter values or regime probabilities are not given, `Estimate()` will calculate them.

MSVAR::DrawErrors

```
DrawErrors(const fAcf);
```

`fAcf`	in:	integer, `TRUE`: draw error analysis

No return value.

Description

Calculates and draws the one-step prediction errors $y_t - \mathsf{E}[y_t|Y_{t-1}]$ and standardized residuals of each equation. If `fAcf` is `TRUE` a graphic analysis of one-step prediction errors and standardized residuals is undertaken. The error analysis includes the estimated ACF, spectral density, histogram and a QQ plot.

MSVAR::DrawFit

```
DrawFit();
```

No return value.

Description

Draws actual and fitted values for all series.

MSVAR::DrawResults

```
DrawResults();
```

Description

Draws the series, the Markov chain component as well as the smoothed, filtered and predicted probabilities for all regimes $m = 1, \ldots, M$.

MSVAR::GetA, MSVAR::GetB

MSVAR::GetMu, MSVAR::GetSigma

MSVAR::GetTrans, MSVAR::GetProbErg

MSVAR::GetProbInit, MSVAR::GetProbLast

MSVAR::GetProbS, MSVAR::GetProbSt

MSVAR::GetProbF, MSVAR::GetProbFt

MSVAR::GetProbP, MSVAR::GetProbPt

MSVAR::GetT, MSVAR::GetU

MSVAR::GetEmOptions, MSVAR::GetModel

MSVAR::GetAIC, MSVAR::GetHQ

MSVAR::GetLogLik, MSVAR::GetSC

Return value

`GetA()`	gets VAR matrices
`GetAIC()`	returns Akaike Information Criterion
`GetB()`	returns $K \times R$ matrix of coefficients (MSxAx: $K \times MR$)
`GetEmOptions()`	returns an array with the EM algorithm options as set using `SetEmOptions`
`GetHQ()`	returns Hannan Quinn Information Criterion
`GetLogLik()`	returns log-likelihood
`GetModel()`	returns an array with the model options as set using `SetModel`
`GetMu()`	returns $K \times M$ matrix of means or intercepts (MSAx: $K \times 1$)
`GetProbInit()`	gets $M \times 1$ vector of initial regime probabilities (MSMx: $M^p \times 1$)
`GetProbErg()`	gets $M \times 1$ vector of ergodic regime probabilities
`GetProbLast()`	gets $N \times 1$ vector of smoothed regime probabilities at time T
`GetProbF()`	gets $N \times T$ matrix of filtered regime probabilities
`GetProbFt()`	gets $M \times T$ matrix of filtered regime probabilities
`GetProbP()`	gets $N \times T$ matrix of predicted regime probabilities
`GetProbPt()`	gets $M \times T$ matrix of predicted regime probabilities
`GetProbS()`	gets $N \times T$ matrix of smoothed regime probabilities
`GetProbSt()`	gets $M \times T$ matrix of smoothed regime probabilities
`GetSC()`	returns Schwarz Information Criterion
`GetSigma()`	returns $K \times K$ variance matrix (MSxxH: $K \times MK$)
`GetT()`	gets number of observations T
`GetTrans()`	returns $M \times M$ transition matrix (transposed matrix of transition probabilities)
`GetU()`	gets $K \times NT$ matrix of residuals

Description

Most of these functions can be only called after the data has been loaded for estimation, or after successful estimation.

MSVAR::MSVAR

```
MSVAR();
```

No return value.

Description

Constructor function.

MSVAR::LoadIn7

MSVAR::LoadDht, MSVAR::LoadFmtVar

MSVAR::LoadObs, MSVAR::LoadVar

MSVAR::LoadWks, MSVAR::LoadXls

```
LoadIn7(const sFilename);
LoadDht(const sFilename, const iYear1, const iPeriod1, const iFreq);
LoadFmtVar(const sFilename);
LoadObs(const sFilename, const cVar,const cObs, const iYear1,
        const iPeriod1, const iFreq, const fOffendMis);
LoadVar(const sFilename, const cVar,const cObs, const iYear1,
        const iPeriod1, const iFreq, const fOffendMis);
LoadWks(const sFilename);
LoadXls(const sFilename);
```

`sFilename`	in:	string, filename
`cVar`	in:	int, number of variables
`cObs`	in:	int, number of observations
`iYear1`	in:	int, start year
`iPeriod1`	in:	int, start period
`iFreq`	in:	int, frequency
`fOffendMis`	in:	int, `TRUE:` offending text treated as missing value `FALSE:` offending text skipped

No return value.

Description

Identical to the functions of the underlying database class:
`LoadDht` creates the database and loads the specified Gauss data file from disk.
`LoadIn7` creates the database and loads the specified GiveWin file (PcGive 7 data file) from disk.
`LoadFmtVar` creates the database and loads the ASCII file with formatting information from disk. In GiveWin this is called 'Data with load info'. Such a file is human-readable, with the data ordered by variable, and each variable preceded by a line of the type:

```
> name year1 period1 year2 period2 frequency
```

`LoadObs` and `LoadVar` create the database and load the specified human-readable data file from disk. The data is ordered by observation (`LoadObs`), or by variable. Since there is no information on the sample or the variable names in these files, the sample must be provided as function arguments. The variable names are set to Var1, Var2, etc., use `Rename` to rename the variables.
`LoadWks` and `LoadXLS` create the database and load the specified spreadsheet file from disk. A .wks or .wk1 file is a Lotus file, an .xls file is an Excel worksheet.

MSVAR::IsConverged

```
IsConverged();
```

Return value

Returns 1 if the EM algorithm converged, 0 otherwise.

MSVAR::LogLik

```
LogLik(const vP, const adFunc, const avScore, const amHess);
```

`vP`	in:	1×1 matrix, with current τ
`adFunc`	in:	address of variable
	out:	loglikelihood at τ
`avScore`	in:	should be 0
`amHess`	in:	should be 0

Return value

Returns 1 if the likelihood can be evaluated, 0 otherwise.

Description

Uses the BHLK filter to evaluate the likelihood.

MSVAR::Select

```
Select(const iGroup, const aSel);
```

`iGroup`	in:	int, group indicator: `Y_VAR`, `X_VAR`, `I_VAR` or `IL_VAR`
`aSel`	in:	array, specifying database name, start lag, end lag

No return value.

Description

Selects variables by name and with specified lags, and assigns the `iGroup` status to the selection. The `aSel` argument is an array consisting of sequences of three values: name, start lag, end lag. For examples, see §9.3. The following types of variables are supported:

`Y_VAR`	dependent and lagged dependent variable
`X_VAR`	exogenous regressors

Each `Select()` adds to the current selection. Use `DeSelect()` to start afresh. *Note:* `SetSample()` checks for data availability; in case of missing observations it uses the largest available sample within the selection.

MSVAR::SetB, MSVAR::SetMu

MSVAR::SetSigma, MSVAR::SetTrans

```
SetB(const mB);
SetMu(const mMu);
SetSigma(const mSigma);
SetTrans(const mTrans);
```

`mMu`	in:	$K \times M$ matrix of means or intercepts (MSAx: $K \times 1$)
`mB`	in:	$K \times R$ matrix of coefficients (MSxAx: $K \times MR$)
`mSigma`	in:	$K \times K$ variance matrix (MSxxH: $K \times MK$)
`mTrans`	in:	$M \times M$ transition matrix (transposed matrix of transition probabilities)

No return value.

Description

Set parameter matrices of the MS-VAR model.

MSVAR::SetEmOptions

```
SetEmOptions(const dTol, const iIt);
SetEmOptions(const dTol, const iIt, const iItMsm);
```

`dTol`	in:	double, tolerance level for convergence of the Em algorithm as percentage change of the log-likelihood (1e-6 by default).
`cItMsm`	in:	integer, maximum number of iterations of the EM algorithm (100 by default).
`cItMsm`	in:	integer, number of internal iterations at each M-step (2 by default).

No return value.

Description

Specifies options of the EM algorithm. Note that the third option only effects MSMx-VAR models.

MSVAR::SetModel

```
SetModel(const fModel, const M);
```

`fModel`	in:	integer, specification of the MS-VAR, see below.
`M`	in:	integer

No return value.

Description

Set the specification of the MS-VAR and the number of regimes to be used in the model. Use `Select()` prior to `SetModel()` to formulate the model.

The following model specifications are supported:

`MSH`	regime-dependent heteroscedasticity
`MSI`	regime-dependent intercept
`MSIH`	regime-dependent intercept and heteroscedasticity
`MSM`	regime-dependent mean
`MSHH`	regime-dependent mean and heteroscedasticity
`MSIA`	regime-dependent intercept
`MSIAH`	regime-dependent intercept and heteroscedasticity
`MSIA`	regime-dependent intercept
`MSIAH`	regime-dependent intercept and heteroscedasticity

Note: The computational burden associated with MSMx-VAR models can be quite high (compared to an MSIx-VAR the factor is M^p where p is the order of the VAR). In general it is not advised to work with a number of regimes $M \geq 4$ due to local maxima and parameter inflation.

MSVAR::SetOptions

```
SetOptions(const fStdErr, const fShowDrawResults, const fSaveDrawWindow);
```

`fStdErr`	in:	integer, `TRUE`: calculate automatically standard errors
`fShowDrawResults`	in:	integer, `TRUE`: calls automatically DrawResults
`fSaveDrawWindow`	in:	integer, `TRUE`: saves gwg files of all MSVAR graphics

No return value.

Description

Sets general options for the MSVAR class.

MSVAR::SetPrint

```
SetPrint(const fPrintResults, const fPrintSteps);
```

`fPrintResults`	in:	int, TRUE or FALSE
`fPrintSteps`	in:	int, TRUE or FALSE

No return value.

Description

Switches printing on (`TRUE`) or off (`FALSE`). By default printing is on. If `fPrintSteps` is `TRUE` the progress of the EM algorithm is printed after each iteration.

MSVAR::SetSample

```
SetSample(const iYear1, const iPeriod1, const iYear2,  const iPeriod2);
```

`iYear1`	in:	integer, start year.
`iPeriod1`	in:	integer, start period.
`iYear2`	in:	integer, end year.
`iPeriod2`	in:	integer, end period.

No return value.

Description

This function selects a subsample in the time dimension. Observations before the specified start sample point and after the end are omitted from estimation. *Note:* `SetSample()` checks for data availability; in case of missing observations it uses the largest available sample within the selection.

MSVAR::StdErr

```
StdErr();
```

No return value.

Description

Prints standard errors based on numerical calculations of the Hessian. If the Hessian is singular the generalized inverse is calculated. As the transition probabilities p_{ij} are restricted to the $[0, 1]$ interval, the parameters are transformed logits $\pi_{ij} = \log\left(\frac{p_{ij}}{1-p_{ij}}\right)$ which avoids problems if one or more of the p_{ij} is close to the border. If one of the transition parameters is estimated to lie on the border, $p_{ij} \in \{0, 1\}$, then the parameter is taken as being fixed and eliminated from the parameter vector (under construction).

Note: The computational burden is proportional to the squared number of parameters. For systems with more than 100 parameters it is suggested to turn the automatical calculation off.