Structural Breaks Estimation for Non-stationary Time Series Models

Richard A. Davis, Thomas C. M. Lee and Gabriel A. Rodriguez-Yam*

Nov, 2004; Revised: May, 2005

Abstract

This paper considers the problem of modeling a class of non-stationary time series using piecewise autoregressive (AR) processes. The number and locations of the piecewise autoregressive segments, as well as the orders of the respective AR processes, are assumed to be unknown. The minimum description length principle is applied to compare various segmented AR fits to the data. The goal is to find the "best" combination of the number of segments, the lengths of the segments, and the orders of the piecewise AR processes. Such a "best" combination is implicitly defined as the optimizer of an objective function, and a genetic algorithm is implemented to solve this difficult optimization problem. Numerical results from simulation experiments and real data analyses show that the procedure enjoys excellent empirical properties. The segmentation of multivariate time series is also considered. Assuming that the true

*Richard Davis is Professor and Chair, Department of Statistics, Colorado State University, Fort Collins, CO. 80523-1877 (email: rdavis@stat.colostate.edu). Thomas Lee is Associate Professor, Department of Statistics, Colorado State University, Fort Collins, CO, 80523-1877 (email: tlee@stat.colostate.edu), and Gabriel Rodriguez-Yam is Postdoctoral fellow, Department of Statistics, Colorado State University, Fort Collins, CO. 80523-1877 (email: rodrigue@stat.colostate.edu). This research was supported in part by NSF grants DMS-0308109 (Davis) and DMS-0203901 (Lee) and by an IBM Faculty Research Award. The authors would like to thank Hernando Ombao for sharing his computer code implementation of Auto-SLEX. They also thank the associate editor and the referee for their constructive comments and suggestions, most of which were incorporated into the final manuscript.

underlying model is a segmented autoregression, this procedure is shown to be consistent for estimating the location of the breaks.

KEY WORDS: Non-stationarity, change-points, minimum description length principle, genetic algorithm.

1 INTRODUCTION

In this paper we consider the problem of modeling a non-stationary time series by segmenting the series into blocks of different autoregressive (AR) processes. The number of break points, denoted by m, as well as their locations and the orders of the respective AR models are assumed to be unknown. We propose an automatic procedure for obtaining such a partition.

In order to describe the setup, for $j = 1, \ldots, m$, denote the break point between the j-th and $(j+1)$-th AR processes as τ_j, and set $\tau_0 = 1$ and $\tau_{m+1} = n + 1$. Then the j-th piece of the series is modeled as an AR process

$$Y_t = X_{t,j}, \quad \tau_{j-1} \leq t < \tau_j, \tag{1}$$

where $\{X_{t,j}\}$ is the AR(p_j) process

$$X_{t,j} = \gamma_j + \phi_{j1} X_{t-1,j} + \ldots + \phi_{j,p_j} X_{t-p_j,j} + \sigma_j \varepsilon_t,$$

$\boldsymbol{\psi}_j := (\gamma_j, \phi_{j1}, \ldots, \phi_{j,p_j}, \sigma_j^2)$ is the parameter vector corresponding to this AR(p_j) process, and the noise sequence $\{\varepsilon_t\}$ is iid with mean 0 and variance 1. Given an observed series $\{y_i\}_{i=1}^n$, the objective is to obtain a "best" fitting model from this class of piecewise AR processes. This is equivalent to finding the "best" combination of the number of pieces $m + 1$, the break point locations $\tau_1, \ldots, \tau_m$, and the AR orders $p_1, \ldots, p_{m+1}$. We propose an automatic procedure for obtaining such a partition. This proposed automatic piecewise autoregressive modeling procedure will be referred to as Auto-PARM. Note that once these parameters are specified, maximum

likelihood estimates of the AR parameters $\boldsymbol{\psi}_j$'s for each segment are easily computed. The primary objective of the methodology developed in this paper is to actually estimate structural breaks for a time series. Under this scenario, it is assumed that some aspect of a time series changes at various times. Such change might be a shift in the mean level of the process, a change in variance, and/or a change in the dependence structure of the process. The sequence of time series between two change-points is assumed to be modeled as a sequence of stationary processes, each of which can be adequately modeled by an autoregressive process. Potential applications of this setup can be found in social sciences in which time series may be impacted by changes in government policies and time series from signal processing engineering, and manufacturing where production processes are often subject to unpredictable changes in the manufacturing process.

As a secondary objective, our methodology can also be viewed as a procedure for approximating locally stationary time series by piecewise AR processes. To see this, we note that the piecewise AR process considered in (1) is a special case of the piecewise stationary process (see also Adak 1998)

$$\tilde{Y}_{t,n} = \sum_{j=1}^{m+1} X_{t,j} I_{[\tau_{j-1}/n, \tau_j/n)}(t/n),$$

where $\{X_{t,j}\}, j = 1, \ldots, m+1$ is a sequence of stationary process. Under certain conditions, Ombao, Raz, Von Sachs, and Malow (2001) argue that locally stationary processes (in the sense of Dahlhaus 1997) can be well approximated by piecewise stationary processes. Roughly speaking, a process is locally stationary if its time-varying spectrum at time t and frequency ω is $|\mathrm{A}(t/n, \omega)|^2$, where $\mathrm{A}(u, \omega)$, $u \in [0, 1]$, $\omega \in [-1/2, 1/2]$ is a continuous function in u. Since AR processes are dense in the class of weakly stationary (purely non-deterministic) processes, the piecewise AR process is *dense* in the class of locally stationary processes.

The above problem of finding a "best" combination of m, τ_j's and p_j's can be treated as a statistical model selection problem, in which candidate models may have different numbers of

parameters. To solve this selection problem we apply the minimum description length (MDL) principle of Rissanen (1989) to *define* a best fitting model (see Saito 1994 and Hansen and Yu 2000 for a comprehensive review of MDL). The basic idea behind the MDL principle is that, the best fitting model is the one that enables the maximum compression of the data. Successes in applying MDL to a variety of practical problems have been widely reported in the literature; see, for example, Lee (2000), Hansen and Yu (2001) and Jornsten and Yu (2003).

As demonstrated below, the best fitted model derived by the MDL principle is defined implicitly as the optimizer of some criterion. Practical optimization of this criterion is not a trivial task, as the search space (consisting of m, τ_j's and p_j's) is enormous. To tackle this problem, we use a genetic algorithm (GA) described for example by Holland (1975). Genetic algorithms are becoming a popular tool in statistical optimization applications (e.g., Gaetan 2000; Pittman 2002; Lee and Wong 2003), and seem particularly well suited for our MDL optimization problem as can be seen in our numerical studies.

Various versions of the above break point detection problem have been considered in the literature. For example, Bai and Perron (1998, 2003) examine the multiple change-point modelling for the case of multiple linear regression, Inclan and Tiao (1994) and Chen and Gupta (1997) consider the problem of detecting multiple variance change-points in a sequence of independent Gaussian random variables, and Kim and Nelson (1999) provide a summary of various applications of the hidden Markov approach to econometrics. Kitagawa and Akaike (1978) implemented an "on-line" procedure based on AIC to determine segments. To implement their method, suppose that an autoregressive model AR(p_0) has been fitted to the dataset $\{y_1, y_2, \ldots, y_{n_0}\}$ and that a new block $\{y_{n_0+1}, \ldots, y_{n_0+n_1}\}$ of n_1 observations becomes available, which can be modeled as an AR(p_1) autoregressive model. Then, the time n_0 is considered a breaking point when the AIC value of the two independent pieces is smaller than the AIC of the autoregressive that results

when the dataset $\{y_1, \ldots, y_{n_0+n_1}\}$ is modeled as a single autoregressive model of order p_2. Each $p_j, j = 0, 1, 2$ is selected among the values $0, 1, \ldots, K$ (K is a predefined value) that minimizes the AIC criterion. The iteration is continued until no more data are available. Like K, n_1 is a predefined value. Ombao et al. (2001) implement a segmentation procedure using the SLEX transformation, a family of orthogonal transformations. For a particular segmentation, a "cost" function is computed as the sum of the costs at all the blocks that define the segmentation. The best segmentation is then defined as the one with minimum cost. Again, because it is not computationally feasible to consider all possible segmentations, they assume that the length of the segments follow a dyadic structure; i.e., an integer power of 2. Bayesian approaches have also been studied; e.g., see Lavielle (1998) and Punskaya et al. (2002). Both procedures choose the final optimal segmentation as the one that maximizes the posterior distribution of the observed series. Numerical results suggest that both procedures enjoy excellent empirical properties. However, theoretical results supporting these procedures are lacking.

For most of the above mentioned procedures, including Auto-PARM, the "best" segmentation is defined as the optimizer of an objective function. Sequential type searching algorithms are adopted by some of these procedures for locating such a "best" segmentation; e.g., Kitagawa and Akaike (1978), Inclan and Tiao (1994) and Ombao et al. (2001). On one hand one would expect that these sequential procedures, when comparing to our genetic algorithm approach, require less computational time to locate a good approximation to the true optimizer. On the other hand, since the genetic algorithm approach examines a much bigger portion of the search space for the optimization, one should also expect that the genetic algorithm approach provides better approximations to the true optimizer. A detailed comparison between the Auto-PARM procedure and the Auto-SLEX procedure of Ombao et al.(2001) is given in Section 4 below.

The rest of this paper is organized as follows. In Section 2 we derive an expression for the MDL

for a given piecewise AR model. In Section 3 we give an overview of the genetic algorithm and discuss its implementation to the segmentation problem. In Section 4 we study the performance of the GA via simulation and in Section 5 the GA is applied to 2 test datasets that have been used in the literature. The case of a multivariate time series and an application is given in Section 6. In Section 7 we summarize our findings and discuss the relative merits of Auto-PARM and other structural break detection procedures. Finally, some theoretical results supporting our procedure are provided in the Appendix.

2 MODEL SELECTION USING MINIMUM DESCRIPTION LENGTH

2.1 Derivation of MDL

This section applies the MDL principle to select a best fitting model from the piecewise AR model class defined by (1). Denote this whole class of piecewise AR models as $\mathcal{M}$ and any model from this class as $\mathcal{F} \in \mathcal{M}$. In the current context the MDL principle *defines* the "best" fitting model from $\mathcal{M}$ as the one that produces the shortest code length that completely describes the observed data $\mathbf{y} = (y_1, y_2, \ldots, y_n)$. Loosely speaking, the code length of an object is the amount of memory space that is required to store the object. In the applications of MDL, one classical way to store $\mathbf{y}$ is to split $\mathbf{y}$ into two components: (i) a fitted model $\hat{\mathcal{F}}$ plus (ii) the portion of $\mathbf{y}$ that is unexplained by $\hat{\mathcal{F}}$. This latter component can be interpreted as the residuals, denoted by $\hat{\mathbf{e}} = \mathbf{y} - \hat{\mathbf{y}}$, where $\hat{\mathbf{y}}$ is the fitted vector for $\mathbf{y}$. If $CL_{\mathcal{F}}(z)$ denotes the code length of object z using model $\mathcal{F}$, one has the following decomposition

$$CL_{\mathcal{F}}(\mathbf{y}) = CL_{\mathcal{F}}(\hat{\mathcal{F}}) + CL_{\mathcal{F}}(\hat{\mathbf{e}}|\hat{\mathcal{F}}),$$

where $CL_{\mathcal{F}}(\hat{\mathcal{F}})$ denotes the code length of the fitted model $\hat{\mathcal{F}}$ and $CL_{\mathcal{F}}(\hat{\mathbf{e}}|\hat{\mathcal{F}})$ is the code length of the corresponding residuals (conditional on the fitted model $\hat{\mathcal{F}}$). In short the MDL principle suggests that a best fitting piecewise AR model $\hat{\mathcal{F}}$ is the one that minimizes $CL_{\mathcal{F}}(\mathbf{y})$.

Now the task is to derive expressions for $CL_{\mathcal{F}}(\hat{\mathcal{F}})$ and $CL_{\mathcal{F}}(\hat{\mathbf{e}}|\hat{\mathcal{F}})$. We begin with $CL_{\mathcal{F}}(\hat{\mathcal{F}})$. Let $n_j := \tau_j - \tau_{j-1}$ denote the number of observations in the j-th segment of $\hat{\mathcal{F}}$. Since $\hat{\mathcal{F}}$ is composed of m, τ_j's, p_j's and $\hat{\boldsymbol{\psi}}_j$'s, we further decompose $CL_{\mathcal{F}}(\hat{\mathcal{F}})$ into

$$\begin{aligned} CL_{\mathcal{F}}(\hat{\mathcal{F}}) &= CL_{\mathcal{F}}(m) + CL_{\mathcal{F}}(\tau_1, \ldots, \tau_m) + CL_{\mathcal{F}}(p_1, \ldots, p_{m+1}) + CL_{\mathcal{F}}(\hat{\boldsymbol{\psi}}_1) + \ldots + CL_{\mathcal{F}}(\hat{\boldsymbol{\psi}}_{m+1}) \\ &= CL_{\mathcal{F}}(m) + CL_{\mathcal{F}}(n_1, \ldots, n_{m+1}) + CL_{\mathcal{F}}(p_1, \ldots, p_{m+1}) + CL_{\mathcal{F}}(\hat{\boldsymbol{\psi}}_1) + \ldots + CL_{\mathcal{F}}(\hat{\boldsymbol{\psi}}_{m+1}). \end{aligned}$$

The last expression was obtained by the fact that complete knowledge of $(\tau_1, \ldots, \tau_m)$ implies complete knowledge of $(n_1, \ldots, n_{m+1})$, and vice versa. In general, to encode an integer I whose value is not bounded, approximately $\log_2 I$ bits are needed. Thus $CL_{\mathcal{F}}(m) = \log_2 m$ and $CL_{\mathcal{F}}(p_j) = \log_2 p_j$. On the other hand, if the upper bound, say I_U, of I is known, approximately $\log_2 I_U$ bits are required. Since all n_j's are bounded by n, $CL_{\mathcal{F}}(n_j) = \log_2 n$ for all j. To calculate $CL_{\mathcal{F}}(\hat{\boldsymbol{\psi}}_j)$, we use the following result of Rissanen: a maximum likelihood estimate of a real parameter computed from N observations can be effectively encoded with $\frac{1}{2}\log_2 N$ bits. Since each of the $p_j + 2$ parameters of $\hat{\boldsymbol{\psi}}_j$ is computed from n_j observations,

$$CL_{\mathcal{F}}(\hat{\boldsymbol{\psi}}_j) = \frac{p_j + 2}{2} \log_2 n_j.$$

Combining these results, we obtain

$$CL_{\mathcal{F}}(\hat{\mathcal{F}}) = \log_2 m + (m+1)\log_2 n + \sum_{j=1}^{m+1} \log_2 p_j + \sum_{j=1}^{m+1} \frac{p_j + 2}{2} \log_2 n_j. \tag{3}$$

Next we derive an expression for $CL_{\mathcal{F}}(\hat{\mathbf{e}}|\hat{\mathcal{F}})$; that is, the code length for the residuals $\hat{\mathbf{e}}$. From Shannon's classical results in information theory, Rissanen demonstrates that the code length of $\hat{\mathbf{e}}$ is given by the negative of the log likelihood of the fitted model $\hat{\mathcal{F}}$. To proceed, let

$\mathbf{y}_j := (y_{\tau_{j-1}}, \ldots, y_{\tau_j - 1})$ be the vector of observations for the j-th piece in (1). For simplicity, we consider that $\boldsymbol{\mu}_j$, the mean of the j-th piece in (1), is $\mathbf{0}$. Denote the covariance matrix of $\mathbf{y}_j$ as $\mathbf{V}_j^{-1} = \text{cov}\{\mathbf{y}_j\}$, and let $\hat{\mathbf{V}}_j$ be an estimate for $\mathbf{V}_j$. Even though the ε_j's are not assumed to be Gaussian, inference procedures will be based on a Gaussian likelihood. Such inference procedures are often referred to *quasi-likelihood*. Assumming the segments are independent, the Gaussian likelihood of the piecewise process is given by

$$L(m, \tau_0, \tau_1, \ldots, \tau_m, p_1, \ldots, p_{m+1}, \boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_{m+1}; \mathbf{y}) = \prod_{j=1}^{m+1} (2\pi)^{-\frac{n_j}{2}} |\mathbf{V}_j|^{\frac{1}{2}} \exp\{-\frac{1}{2}\mathbf{y}_j^T \mathbf{V}_j \mathbf{y}_j\},$$

and hence the code length of $\hat{\mathbf{e}}$ given the fitted model $\hat{\mathcal{F}}$ is

$$\begin{aligned} CL_{\mathcal{F}}(\hat{\mathbf{e}}|\hat{\mathcal{F}}) &\approx -\log_2 L(m, \tau_0, \tau_1, \ldots, \tau_m, \hat{\boldsymbol{\psi}}_1, \ldots, \hat{\boldsymbol{\psi}}_{m+1}; \mathbf{y}) \\ &= \sum_{j=1}^{m+1} \{\frac{n_j}{2}\log(2\pi) - \frac{1}{2}\log|\hat{\mathbf{V}}_j| + \frac{1}{2}\mathbf{y}_j^T \hat{\mathbf{V}}_j \mathbf{y}_j\} \log_2 e. \end{aligned} \tag{4}$$

Combining (3) and (4) and using logarithm base e rather than base 2, we obtain the following approximation for $CL_{\mathcal{F}}(\mathbf{y})$ by

$$\begin{aligned} \log m + (m+1)\log n + \sum_{j=1}^{m+1} \log p_j + \sum_{j=1}^{m+1} \frac{p_j + 2}{2} \log n_j \\ + \sum_{j=1}^{m+1} \{\frac{n_j}{2}\log(2\pi) - \frac{1}{2}\log|\hat{\mathbf{V}}_j| + \frac{1}{2}\mathbf{y}_j^T \hat{\mathbf{V}}_j \mathbf{y}_j\}. \end{aligned} \tag{5}$$

Using the standard approximation to the likelihood for AR models, i.e., $-2\log(\text{likelihood})$ by $n_j \log \hat{\sigma}_j^2$, where $\hat{\sigma}_j^2$ is the Y-W estimate of σ_j^2 (Brockwell and Davis 1991), we define

$$\begin{aligned} \text{MDL}(m, \tau_1, \ldots, \tau_m, p_1, \ldots, p_{m+1}) = \log m + (m+1)\log n + \sum_{j=1}^{m+1} \log p_j \\ + \sum_{j=1}^{m+1} \frac{p_j + 2}{2} \log n_j + \sum_{j=1}^{m+1} \frac{n_j}{2} \log(2\pi\hat{\sigma}_j^2) \end{aligned} \tag{6}$$

We propose selecting the best fitting model for $\mathbf{y}$ as the model $\mathcal{F} \in \mathcal{M}$ that minimizes $\text{MDL}(m, \tau_1, \ldots, \tau_m, p_1, \ldots, p_{m+1})$.

2.2 Consistency

To this point, we have not assumed the existence of a true model for the time series. However, to study theoretical properties of these estimates, an underlying model must be specified. Here we assume that there exist true values m_0 and λ_j^0, $j = 1, \ldots, m_0$, such that $0 < \lambda_1^0 < \lambda_2^0 < \cdots < \lambda_{m_0}^0 < 1$. The observations $y_1, \ldots, y_n$ are assumed to be a realization from the piecewise AR process defined in (1) with $\tau_i = [\lambda_i^0 n]$, $i = 1, 2, \ldots, m_0$, where $[x]$ is the greatest integer that is less than or equal to x. In estimating the break points $\tau_1, \ldots, \tau_{m_0}$, it is necessary to require that the segments have a sufficient number of observations to adequately estimate the specified AR parameter values. Otherwise, the estimation is over-determined resulting in an infinite value for the likelihood. So, to ensure sufficient separation of the breakpoints, choose $\epsilon > 0$ small such that $\epsilon \ll \min_{i=1,\ldots,m_0+1}(\lambda_i^0 - \lambda_{i-1}^0)$ and set

$$A_m = \{(\lambda_1, \ldots, \lambda_m), 0 < \lambda_1 < \lambda_2 < \ldots < \lambda_m < 1, \lambda_i - \lambda_{i-1} \geq \epsilon, i = 1, 2, \ldots, m+1\},$$

where $\lambda_0 := 0$ and $\lambda_{m+1} := 1$. Setting $\lambda := (\lambda_1, \ldots, \lambda_m)$ and $p = (p_1, \ldots, p_{m+1})$, the parameters m, λ and p are then estimated by minimizing MDL over $m \leq M_0$, $0 \leq p \leq P_0$, and $\lambda \in A_m$., i.e,

$$\hat{m}, \hat{\lambda}, \hat{p} = \arg \min_{\substack{m \leq M_0, 0 \leq p \leq P_0 \\ \lambda \in A_m}} \frac{2}{n} \mathrm{MDL}(m, \lambda, p),$$

where M_0 and P_0 are upper bounds for m and p_j, respectively. In the appendix we prove the following consistency result.

Proposition 1 *For the model specified in (1), when m_0, the number of break points is known, then $\hat{\lambda}_j \to \lambda_j^0$, a.s., $j = 1, 2, \ldots, m_0$.*

In Proposition 1, the true number of breaks m_0 is assumed known. As the simulation studies in Section 4 show, for unknown m_0, the estimator $\hat{m}_0$ obtained with our procedure seems to be

consistent, although we do not have a proof. Even in the independent case, the consistency of $\hat{m}_0$ is known in only some special cases (e.g., Lee 1997 and Yao 1988).

3 OPTIMIZATION USING GENETIC ALGORITHMS

As the search space is enormous, optimization of $\mathrm{MDL}(m, \tau_1, \ldots, \tau_m, p_1, \ldots, p_{m+1})$ is a nontrivial task. In this section we propose using a genetic algorithm (GA) to effectively tackle this problem.

3.1 General Description

The basic idea of the canonical form of GAs can be described as follows. An initial set, or population, of possible solutions to an optimization problem is obtained and represented in vector form. These vectors are often called *chromosomes* and are free to "evolve" in the following way. Parent chromosomes are randomly chosen from the initial population and chromosomes having lower (higher) values of the objective criterion to be minimized (maximized) would have a higher chance of being chosen. Then offspring are produced by applying a *crossover* or a *mutation* operation to the chosen parents. Once a sufficient number of such second generation offspring are produced, third generation offspring are further produced from these second generation offspring in a similar fashion. This process continues for a number of generations. If one believes in Darwin's *Theory of Natural Selection*, the expectation is that objective criterion values of the offspring will gradually improve over generations and approach the optimal value.

In a crossover operation, one child chromosome is produced from "mixing" two parent chromosomes. The aim is to allow the possibility that the child receives different best parts from its parents. A typical "mixing" strategy is that every child gene location has an equal chance of receiving either the corresponding father gene or the corresponding mother gene. This crossover operation is the distinct feature that makes genetic algorithms different from other optimization

methods. For possible variants of the crossover operation, consult Davis (1991).

In a mutation operation one child chromosome is produced from one parent chromosome. The child is essentially the same as its parent except for a small number of genes where randomness is introduced to alter the types of genes. Such a mutation operation prevents the algorithm from being trapped in local optima.

In order to preserve the best chromosome of a current generation, an additional step, called the *elitist* step, may be performed. Here the worst chromosome of the next generation is replaced with the best chromosome of the current generation. Inclusion of this elitist step guarantees the monotonicity of the algorithm.

There are many variations of the above canonical GA. For example, parallel implementations can be applied to speed up the convergence rate as well as to reduce the chance of converging to sub-optimal solutions (Forrest 1991; Alba and Troya 1999). In this paper we implement the *Island Model*. Instead of running only one search in one giant population, the island model simultaneously runs NI (Number-of-Islands) canonical GAs in NI different sub-populations. The key feature is, periodically, a number of individuals are migrated amongst the islands according to some migration policy. The migration can be implemented in numerous ways (Martin, Lienig and Cohoon 2000; Alba and Troya 2002). In this paper, we adopt the following migration policy: after every M_i generations, the worst M_N chromosomes from the j-th island are replaced by the best M_N chromosomes from the $(j-1)$-th island, $j = 1, \ldots, NI$. For $j = 1$ the best M_N chromosomes are migrated from the NI-th island. In our simulations we used $NI = 40$, $M_i = 5$, $M_N = 2$ and a sub-population size of 40.

3.2 Implementation Details

This subsection provides details of our implementation of the GAs that is tailored to our piecewise AR model fitting.

Chromosome Representation: The performance of a genetic algorithm certainly depends on how a possible solution is represented as a chromosome, and for the current problem a chromosome should carry complete information for any $\mathcal{F} \in \mathcal{M}$. That is, the break points τ_j's as well as the AR orders p_j's. Once these quantities are specified, maximum likelihood estimates of other model parameters can be uniquely determined. Here we propose using the following chromosome representation: a chromosome $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_n)$ is of length n with gene values δ_t defined as

$$\delta_t = \begin{cases} -1, & \text{if no break point at } t, \\ p_j, & \text{if } t = \tau_{j-1} \text{ and the AR order for the } j\text{-th piece is } p_j. \end{cases}$$

Furthermore, the following "minimum span" constraint is imposed on $\boldsymbol{\delta}$: say if the AR order of a certain piece in $\mathcal{F}$ is p, then this piece is made to have at least m_p observations. This predefined integer m_p is chosen to guarantee that there are enough observations for obtaining quality estimates for the parameters of the AR(p) process. Also, in the practical implementation of the algorithm, one needs to impose an upper bound P_0 on the order p_j's of the AR processes. There seems to be no universal choice for P_0, as for complicated series one needs a large P_0 to capture for example seasonality, while for small series P_0 cannot be larger than the number of observations n. For all our numerical works we set $P_0 = 20$, and the corresponding minimum span m_p's are listed in Table 1.

Table 1: Values of m_p used in the simulations.

p	0-1	2	3	4	5	6	7-10	11-20
m_p	10	12	14	16	18	20	25	50

Our empirical experience suggests that the above representation scheme, together with the minimum span constraint, is extremely effective for the purpose of using GAs to minimize

MDL$(m, \tau_1, \ldots, \tau_m, p_1, \ldots, p_{m+1})$. It is most likely due to the fact that the location information of the break points and the order of the AR processes are explicitly represented.

Initial Population Generation: Our implementation of the GA starts with an initial population of chromosomes generated at random. For this procedure, the user value π_B, the probability that the "j-th location" of the chromosome being generated be a break point is needed. A large value of π_B makes the initial chromosomes to have a large number of break points, thus a small value is preferred. We use $\pi_B = \min(m_p)/n = 10/n$ (in Section 4 a sensitivity analysis for this parameter is given). Once a location is declared to be a break, an AR order is selected from the uniform distribution with values 0, 1,..., P_0. The following strategy was used to generate each initial chromosome. First, select a value for p_1 from $\{0, \ldots, P_0\}$ with equal probabilities and set $\delta_1 = p_1$; i.e., the first AR piece is of order p_1. Then the next $m_{p_1} - 1$ genes δ_i's (i.e., δ_2 to $\delta_{m_{p_1}}$) are set to -1, so that the above minimum span constraint is imposed for this first piece. Now for the next gene $\delta_{m_{p_1}+1}$ in line. It will either be initialized as a break point (i.e., assigned a non-negative integer p_2) with probability π_B, or it will be assigned -1 with probability $1 - \pi_B$. If it is to be initialized as a break point, then we set $\delta_{m_{p_1}+1} = p_2$, where p_2 is randomly drawn from $\{0, \ldots, P_0\}$. This implies that the second AR process is of order p_2, and the next $m_{p_2} - 1$ δ_i's will be assigned -1 so that the minimum span constraint is guaranteed. On the other hand, if $\delta_{m_{p_1}+1}$ is to be assigned with -1, the initialization process will move to the next gene in line and decide if this gene should be a break point gene or a "-1" gene. This process continues in a similar fashion, and a random chromosome is generated when the process hits the last gene δ_n.

Crossover and Mutation: Once a set of initial random chromosomes is generated, new chromosomes are generated by either a crossover or a mutation operation. In our implementation we set the probability for conducting a crossover operation as $\pi_C = 1 - \min(m_p)/n = (n - 10)/n$.

For the crossover operation, two parent chromosomes are chosen from the current population

of chromosomes. These two parents are chosen with probabilities inversely proportional to their ranks sorted by their MDL values. In other words, chromosomes that have smaller MDL values will have higher chances to be selected. From these two parents, the gene values δ_i's of the child chromosome will be inherited in the following manner. Firstly for $t = 1$, δ_t will take on the corresponding δ_t value from either the first or the second parent with equal probabilities. If this value is -1, then the same gene–inheriting process will be repeated for the next gene in line (i.e., δ_{t+1}). If this value is not -1, then it is a non-negative integer p_j denoting the AR order of the current piece. In this case the minimum span constraint will be imposed (i.e., the next $m_{p_j} - 1$ δ_t's will be set to -1), and the same gene–inheriting process will be applied to the next available δ_t.

For mutation one child is reproduced from one parent. Again, this process starts with $t = 1$, and every δ_t (subject to the minimum span constraint) can take on one of the following three possible values: (i) with probability π_P it will take the corresponding δ_t value from the parent, (ii) with probability π_N it will take the value -1, and (iii) with probability $1 - \pi_P - \pi_N$, it will take the a new randomly generated AR order p_j. In this paper we set $\pi_P = 0.3$ and $\pi_N = 0.3$.

Declaration of Convergence: Recall that we adopt the Island Model in which migration is allowed for every $M_i = 5$ generations. At the end of each migration the overall best chromosome (i.e., the chromosome with smallest MDL) is noted. If this best chromosome does not change for 10 consecutive migrations, or the total number of migrations exceeds 20, this best chromosome is taken as the solution to this optimization problem.

4 SIMULATION RESULTS

Five sets of simulation experiments were conducted to evaluate the practical performances of Auto-PARM. The experimental setups of the first two simulations are taken from Ombao et al. (2001),

for which the authors use them to test their Auto-SLEX procedure. In the first simulation, the pieces of the true process follow a dyadic structure; i.e., the length of each segment is a integer power of 2. In the second and fourth simulations the true process does not contain any structural breaks, but its time-varying spectrum changes very slowly over time. In the third simulation the process contains three pieces, one of which is an ARMA(1,1) process and another is a MA(1) process. In the last simulation the process has two distinctive features: the pieces do not follow a dyadic structure and the length of one of the pieces is very short.

For the results reported in this section and in Section 5, we obtained slightly better results by minimizing MDL based on the exact likelihood function evaluated at Yule-Walker estimates. That is, MDL as defined by (5) was used in all of the simulation results in this section. Throughout the whole section, results reported for Auto-SLEX were obtained using computer code provided by Dr. Hernando Ombao.

4.1 Piecewise Stationary Process with Dyadic Structure

In this simulation example, the target non-stationary series is generated with the following model

$$Y_t = \begin{cases} 0.9Y_{t-1} + \varepsilon_t, & \text{if } 1 \le t \le 512, \\ 1.69Y_{t-1} - 0.81Y_{t-2} + \varepsilon_t, & \text{if } 513 \le t \le 768, \\ 1.32Y_{t-1} - 0.81Y_{t-2} + \varepsilon_t, & \text{if } 769 \le t \le 1024, \end{cases} \tag{9}$$

where $\varepsilon_t \sim$ iid $\mathrm{N}(0,1)$. The main feature of this model is that the lengths of the pieces are a power of 2. This is in fact ideally suited for the Auto-SLEX procedure of Ombao et al. (2001). A typical realization of this process is shown in Figure 1. For $\omega \in [0, 0.5)$, let $f_j(\omega)$ be the spectrum of the j-th piece, i.e.,

$$f_j(\omega) = \sigma_j^2 |1 - \phi_{j1} \exp\{-i2\pi\omega\} - \ldots - \phi_{jp_j} \exp\{-i2\pi p_j \omega\}|^2, \tag{10}$$

then for $t \in [\tau_{j-1},\ \tau_j)$, the *time-varying spectrum* of the process Y_t in (1) is $f(t/n, \omega) = f_j(\omega)$. The true spectrum of the process in (9) is shown in the middle panel of Figure 2, where darker shades represent higher power.

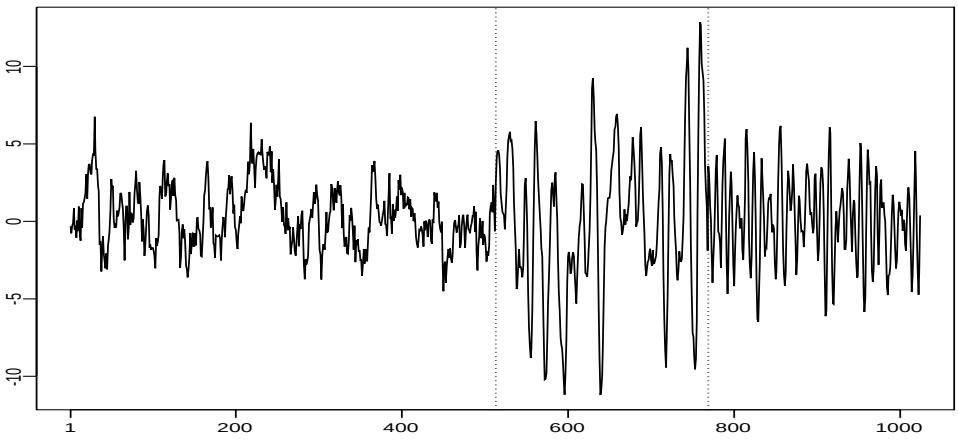

Figure 1: A realization from the piecewise stationary process in (9).

We applied Auto-PARM to the realization in Figure 1 and obtained two break points located at $\hat{\tau}_1 = 512$ and $\hat{\tau}_2 = 769$, indicated by the dotted vertical lines in the figure. The Auto-PARM correctly identified the AR orders ($\hat{p_1}$=1, $\hat{p_2}$=2 and $\hat{p_3}$=2) for this realization. From this segmentation, the time varying spectrum of this realization was estimated as $\hat{f}_{t/n}(\omega) = \hat{f}_j(\omega)$, where $\hat{f}_j(\omega)$ is obtained by replacing parameters in (10) with their corresponding estimates . The estimated time varying spectrum is displayed in the left panel of Figure 2. Our implementation of Auto-PARM, which is written in Compaq Visual Fortran, took 2.34 seconds on a 1.6 Ghz intel pentium M processor to obtain the above estimates. The Auto-SLEX time varying spectrum of this realization is shown in the panel on the right of Figure 2.

Next, 200 realizations of the process in (9) were simulated and Auto-PARM was applied to segment each of these realizations. Table 2 lists the percentages of the fitted number of segments. For comparative purposes, the corresponding values of the Auto-SLEX method are also listed. Notice that Auto-PARM gave the correct number of segments for 96% of the 200 realizations,

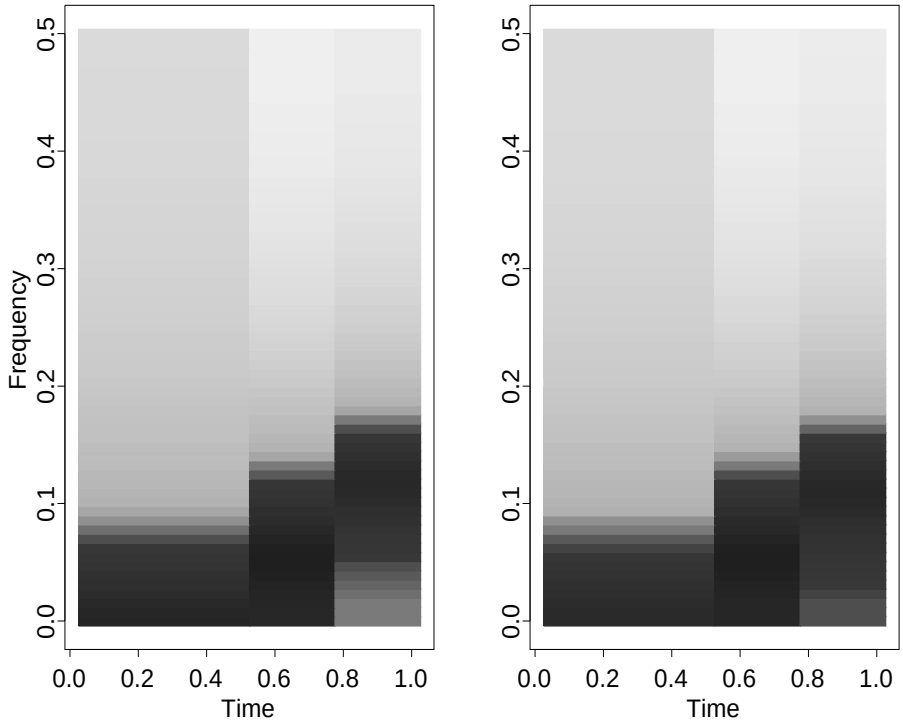

Figure 2: True time-varying log-spectrum of process in (9) and Auto-PARM and Auto-SLEX estimates from the realization of Figure 1.

while Auto-SLEX gave the correct segmentation for 73% of the realizations. Table 2 also reports, for each $\hat{m}$, the mean and standard deviation of $\hat{\lambda}_j := (\hat{\tau}_j - 1)/n$, $j = 2, \ldots, \hat{m}$, where $\hat{\tau}_j$ is the Auto-PARM estimate of τ_j. For convenience we will refer to $\hat{\lambda}_j$ as a *relative* break point.

Table 3 lists the relative frequencies of the AR order p estimated by the Auto-PARM procedure for the 96% of the realizations with 3 pieces. Of the 200 realizations, 44% have two breaks and AR orders 1, 2 and 2, respectively. For these realizations, the means and the standard errors of the estimated parameters $\phi_1, \ldots, \phi_{p_j}$, σ_j^2 are shown in Table 4. From these tables one can see that the practical performance of Auto-PARM applied to the above piecewise stationary process performs extremely well, especially for locating the break points.

Sensitivity Analysis

We also considered the sensitivity of the GA to the probabilities of initialization (π_B) and crossover

Table 2: Summary of the estimated break points from both the Auto-SLEX and Auto-PARM procedures for the process (9). For Auto-PARM the means and standard errors of the relative break points are also reported.

Number of segments	Auto-SLEX break points (%)	ASE	Auto-PARM break points (%)	mean	std	ASE
2	2.5	0.396	0.0			
		(0.019)				
3	73.0	0.121	96.0	0.500	0.007	0.049
		(0.027)		0.750	0.005	(0.030)
4	11.0	0.146	4.0	0.496	0.004	0.140
		(0.040)		0.566	0.108	(0.036)
				0.752	0.003	
5	9.5	0.206	0.0			
		(0.045)				
≥ 6	4.0	0.253	0.0			
		(0.103)				
All	100.0	0.144	100.0			0.052
		(0.064)				(0.035)

Table 3: Relative frequencies of the AR order estimated by the Auto-PARM procedure for the realizations of model (9).

Order	0	1	2	3	4	5	6	≥ 7
p_1	0	99.0	1.0	0	0	0		
p_2	0	0	67.7	16.7	9.9	3.6	0.5	1.5
p_3	0	0	60.4	22.9	5.7	6.8	2.1	2.1

(π_C). To assess the sensitivity, Auto-PARM was applied to the same realizations used in Table 2 for each combination of values of $\pi_B \in \{0.01, 0.1\}$ and $\pi_C = \{0.90, 0.99\}$. The others parameter values in the implementation of Auto-PARM are as described in Section 3.

The relative frequency of the number of break points estimated by Auto-PARM is shown in Table 5 (columns 4 and 5). For the replicates with 3 pieces, the mean of the break points and standard errors are shown in columns 6 and 7, respectively. The frequency of the correct AR order estimated by Auto-PARM for each piece is shown in columns 8, 9 and 10. The average of the MDL values and the standard error are shown in the last two columns. The column labeled

Table 4: Summary of parameter estimates obtained by Auto-PARM for the realizations that have 2 breaks and pieces with orders 1, 2 and 2, respectively. For each segment, the true parameters, the mean and the standard errors (in parenthesis) are shown.

			Parameter		
Segment	Model		ϕ_1	ϕ_2	σ^2
I	AR(1)	true	0.90		1.00
		mean	0.89		1.02
			(0.02)		(0.07)
II	AR(2)	true	1.69	-0.81	1.00
		mean	1.65	-0.78	1.12
			(0.05)	(0.05)	(0.19)
III	AR(2)	true	1.32	-0.81	1.00
		mean	1.30	-0.79	1.07
			(0.04)	(0.04)	(0.13)

time shows the average time in seconds to implement Auto-PARM.

Table 5: Sensitivity analysis ($NI\times$ popsize = 40 × 40). Summary of sensitivity analysis of π_B and π_C of Auto-PARM based on 200 realizations of (9).

			Number of breaks (%)		Auto-PARM Break points		AR order $\hat{p}_1$	$\hat{p}_2$	$\hat{p}_3$	
π_B	π_C	time	2	3	mean	std	1	2	2	MDL
0.01	0.90	14.97	91.5	8.5	0.500	0.008	99.5	57.4	60.1	1520.45
					0.749	0.007				
0.01	0.99	3.0	95.5	4.5	0.499	0.009	99.5	56.5	60.2	1520.56
					0.750	0.007				
0.10	0.90	16.85	95.5	4.5	0.499	0.010	98.4	53.4	53.4	1519.41
					0.750	0.008				
0.10	0.99	4.9	94.5	5.5	0.499	0.008	97.9	57.1	57.1	1519.22
					0.750	0.007				

From Table 5, we see that distinct values of π_B and π_C give comparable values of MDL. Notice that Auto-PARM runs the fastest for the values selected for π_B and π_C in Section 3, i.e. $\pi_B = \min(m_p)/n$ and $\pi_C = 1 - \min(m_p)/n$. As seen from this table, there is little impact on the choice of initial values for π_B and π_C in executing Auto-PARM.

4.2 Slowly Varying AR(2) Process

The true model considered in this second simulation experiment does not possess a structural break. Rather, the process has a slowly changing spectrum given by the following time-dependent AR(2) model

$$Y_t = a_t Y_{t-1} - 0.81 Y_{t-2} + \varepsilon_t, \quad t = 1, 2, \ldots, 1024, \tag{11}$$

where $a_t = 0.8\{1 - 0.5\cos(\pi t/1024)\}$ and $\varepsilon_t \sim$ iid N$(0, 1)$. A typical realization of this process is shown in Figure 3, while the spectrum of this process is shown in the middle panel of Figure 5.

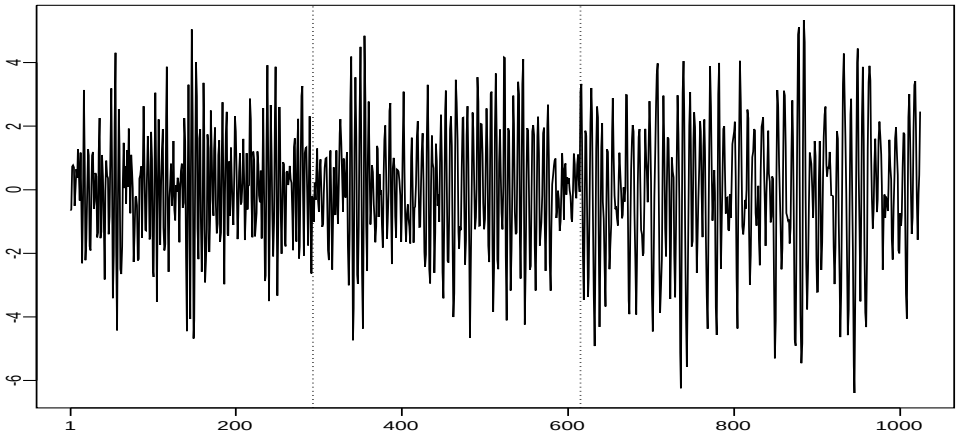

Figure 3: Realization from the process in (11).

For the realization in Figure 3, the Auto-PARM procedure segmented the process into three pieces with break points located at $\hat{\tau}_1 = 318$ and $\hat{\tau}_2 = 614$ (vertical dotted lines in this figure). Also, each of the three pieces was modeled as an AR(2) process. The run time for this fitting was 1.79 seconds. Based on the model found by Auto-PARM, the time-varying spectrum of this realization was computed and is shown in the left panel of Figure 4. Also, the Auto-SLEX time-varying spectrum of this realization is shown in the right panel of this figure.

Next we generated 200 realizations of the above process, and the corresponding Auto-PARM estimates were obtained. Since there are no true structural breaks in such realizations, we follow Ombao et al. (2001) and use the average squared error as a numerical error measure of perfor-

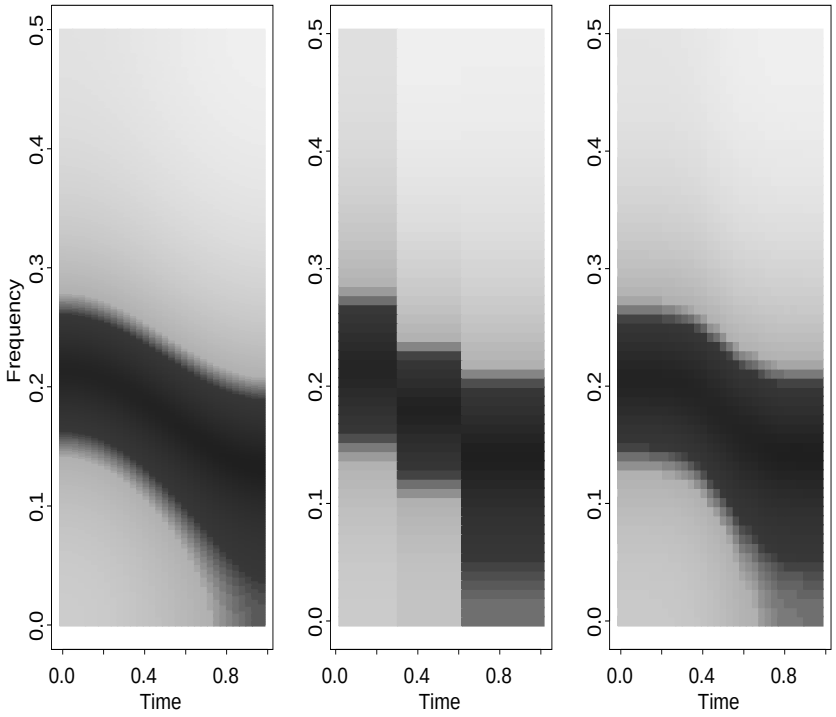

Figure 4: Auto-PARM and Auto-SLEX estimates of log-spectrum of process in (11) for the realization from Figure 3.

mance. The ASE is defined by

$$\text{ASE} = \{n(M_J/2+1)\}^{-1} \sum_{t=1}^{n} \sum_{k=0}^{M_J/2} \{\log \hat{f}(t/n, \omega_k) - \log f(t/n, \omega_k)\}^2,$$

where $\hat{f}(\cdot,\cdot)$ is an estimate of the true time-dependent spectrum $f(\cdot,\cdot)$ of the process, J is a pre-specified scale satisfying $J < L = \log_2(n)$ and $M_J := n/2^J$ (see equation (19) in Ombao et al. 2001). In this simulation we took $J = 4$.

The number of segments, locations of the break points and the ASEs of the Auto-PARM estimates are summarized in Table 6. Also listed in Table 6 are the ASE values of the Auto-SLEX procedure. From Table 6 the following two main observations can be made. First, for each of the simulated processes, Auto-PARM produces either two or three segments that are of roughly the same length, while the Auto-SLEX procedure tends to split the process into a larger number of segments. Second, the ASE values of Auto-PARM are smaller than those from Auto-SLEX.

In order to show a "consistency" like property of Auto-PARM, we computed the average of

Table 6: Break points and ASEs values from the Auto-PARM and the Auto-SLEX estimates computed from 200 realizations of (11). Numbers inside parentheses are standard errors of the ASE values.

Number of segments	Auto-SLEX (%)	Auto-SLEX ASE	Auto-PARM (%)	Auto-PARM break points mean	Auto-PARM break points std	Auto-PARM ASE
1	0.0	-	0.0	-	-	-
2	40.5	0.191	37.5	0.496	0.055	0.129
		(0.019)				(0.015)
3	37.0	0.171	62.0	0.365	0.074	0.081
		(0.022)		0.662	0.079	(0.016)
4	15.0	0.174	0.5	0.308	-	0.10
		(0.029)		0.538	-	-
				0.875	-	
5	5.0	0.202				
		(0.045)				
≥ 6	2.5	0.223				
		(0.037)				
All	100.0	0.182	100.0			0.099
		(0.027)				(0.028)

all the time-varying spectra of the 200 Auto-PARM and Auto-SLEX estimates. The averaged Auto-PARM spectrum is displayed in the left panel of Figure 5 and looks remarkably similar to the true time varying spectrum. Also the averaged Auto-SLEX spectrum is shown in the right panel of this figure. Lastly in Table 7 we summarize the Auto-PARM estimates of the AR orders for the above process. Notice that most of the segments were modeled as AR(2) processes.

Table 7: Relative frequencies of the AR order selected by Auto-PARM for the realizations from the process (11).

Order	0	1	2	3	4	≥ 5
			2-segment realizations			
p_1	0	0	97.3	1.3	1.3	0
p_2	0	0	93.3	5.3	1.3	0
			3-segment realizations			
p_1	0	0	100.0	0	0	0
p_2	0	0	94.4	4.8	0.8	0
p_3	0	0	91.1	8.1	0.8	0

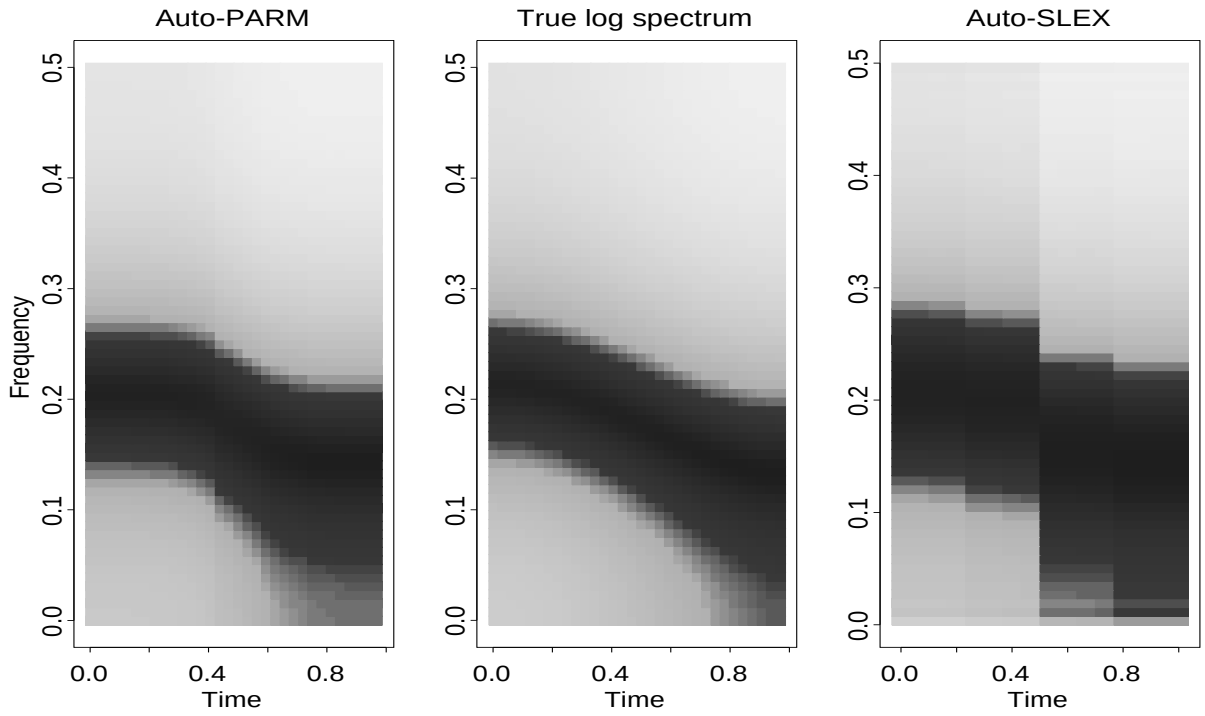

Figure 5: True time-varying log-spectrum of process in (11) (center) and Auto-PARM and Auto-SLEX log-spectrum estimate (average of log-spectrum estimates obtained from 200 realizations).

4.3 Piecewise ARMA process

Recall that the Auto-PARM procedure assumes the observed process is composed of a series of stationary AR processes. This third simulation, designed to assess the performance of Auto-PARM when the AR assumption is violated, has data generating model given by

$$Y_t = \begin{cases} -0.9Y_{t-1} + \varepsilon_t + 0.7\varepsilon_{t-1}, & \text{if } 1 \le t \le 512, \\ 0.9Y_{t-1} + \varepsilon_t, & \text{if } 513 \le t \le 768, \\ \varepsilon_t - 0.7\varepsilon_{t-1}, & \text{if } 769 \le t \le 1024, \end{cases} \tag{12}$$

where $\varepsilon_t \sim$ iid N$(0, 1)$. Notice that the first piece is an ARMA(1,1) process while the last piece is a MA(1) process. A typical realization of this process is shown in Figure 6.

The Auto-PARM procedure was applied to the realization in Figure 6. Three pieces were obtained. The break points are at $\hat{\tau}_1 = 513$ and $\hat{\tau}_2 = 769$ (dotted vertical lines in this figure),

Table 8: Summary of parameter estimates of slowly varying AR(2) process realizations segmented by Auto-PARM as two and three pieces, where each piece is an AR(2) process. For each segment, the true parameters, their mean and standard deviation (in parenthesis) are shown.

		Parameter		
j-th piece		ϕ_1	ϕ_2	σ^2
2-piece realizations with AR(2) pieces: 68				
1	true		-0.81	1.00
	mean	0.54	-0.79	1.05
	std	(0.04)	(0.03)	(0.07)
2	true		-0.81	1.00
	mean	1.05	-0.79	1.05
	std	(0.04)	(0.03)	(0.07)
3-piece realizations with AR(2) pieces: 106				
1	true		-0.81	1.00
	mean	0.46	-0.80	1.03
	std	(0.06)	(0.03)	(0.08)
2	true		-0.81	1.00
	mean	0.82	-0.81	1.01
	std	(0.08)	(0.04)	(0.10)
2	true		-0.81	1.00
	mean	1.14	-0.80	1.06
	std	(0.05)	(0.04)	0.10

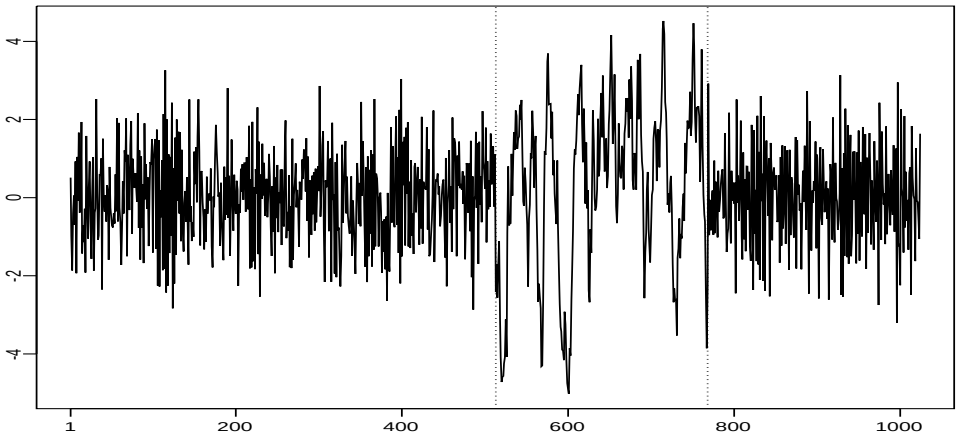

Figure 6: A realization from the piecewise stationary process in (12).

while the order of the AR processes are 4, 1 and 2 respectively. The total run time for this fit was 1.53 seconds. The time-varying spectrum (not shown here) based on the model found by Auto-PARAM is reasonable close to the true spectrum (not shown here) even though two of the segments are not AR processes.

To assess the large sample behavior of Auto-PARM, 200 realizations from (12) were generated,

and the corresponding Auto-PARM estimates were obtained. An encouraging result is that for all 200 realizations, Auto-PARM always gave the correct number of stationary segments. The estimates of the break point locations are summarized in Table 9. In Table 10 we show the relative frequency of the AR order p_j selected to model the pieces of the realizations. As expected, quite often large AR orders were selected for the ARMA and MA segments.

Table 9: Summary of Auto-PARM estimated break points obtained from 200 realizations from the process in (12).

Number of segments	%	relative break points mean	std
3	100.0	0.50	0.005
		0.75	0.003

Table 10: Relative frequencies of the AR order selected by Auto-PARM for the realizations from the process (12).

Order	0	1	2	3	4	5	6	7	≥ 8
p_1	0	4.0	22.5	40.0	23.5	8.5	1.0	0.5	0
p_2	0	89.5	8.5	1.5	0.5	0	0	0	0
p_3	0	0.5	22.0	45.0	19.5	7.5	4.5	1.0	0

4.4 Time varying MA(2) process

Like the example in Section 4.2, the true model considered in this last simulation experiment does not possess a structural break. Rather, the process has a changing spectrum given by the following time-dependent MA(2) model

$$Y_t = \varepsilon_t + a_t \varepsilon_{t-1} + 0.5\varepsilon_{t-2}, \quad t = 1, 2, \ldots, 1024, \tag{13}$$

where $a_t = 1.122\{1 - 1.781 \sin(\pi t/2048)\}$ and $\varepsilon_t \sim$ iid N$(0, 1)$. A typical realization of this process is shown in Figure 7, while the spectrum of this process is shown on the left panel of Figure 9.

For the realization in Figure 7, the Auto-PARM procedure segmented it into four AR pieces of orders 5, 3, 5, and 3, respectively with break points located at $\hat{\tau}_1 = 109$, $\hat{\tau}_2 = 307$, and $\hat{\tau}_3 = 712$ (vertical dotted lines in this figure). The run time for this model fit was 3.76 seconds. Based on

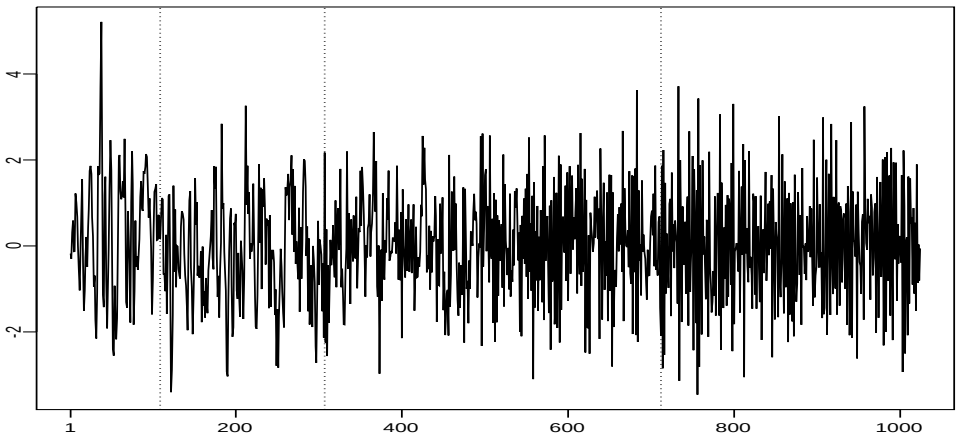

Figure 7: Realization from the process in (13).

the model found by Auto-PARM, the time-varying spectrum of this realization is shown in the left panel of Figure 8. For comparison, the Auto-SLEX time-varying spectrum estimate of this realization is shown in the right panel of this figure.

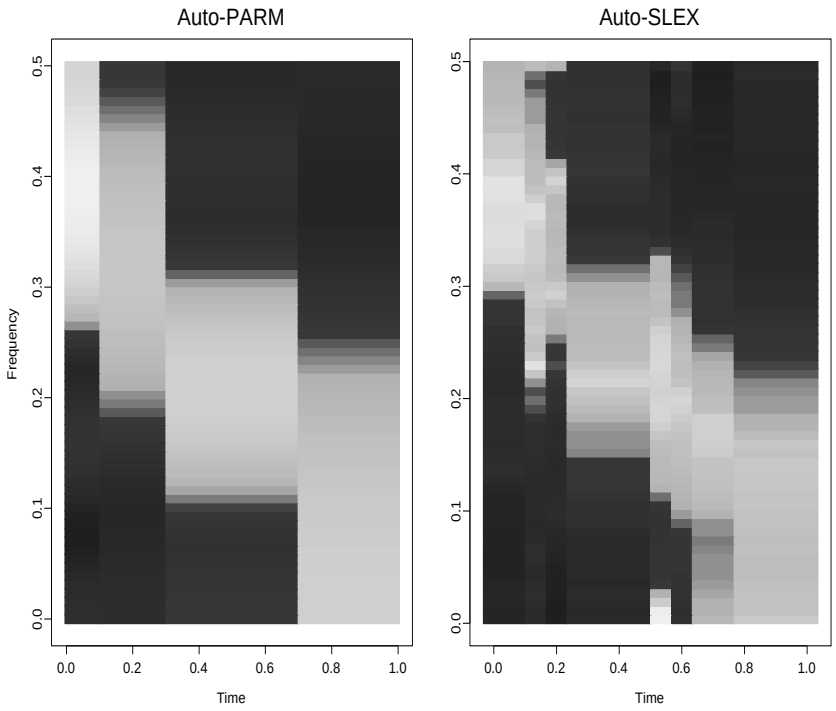

Figure 8: Auto-PARM and Auto-SLEX estimates of log-spectrum of process in (13) for the realization from Figure 7.

Next we generated 200 realizations of the above process, and the corresponding Auto-PARM

Table 11: Summary of the estimated break points from both the Auto-SLEX and Auto-PARM procedures for the process (13). For Auto-PARM the means and standard errors of the relative break points are also reported. Numbers inside parentheses are standard errors of the ASE values.

Number of segments	Auto-SLEX break points (%)	ASE	Auto-PARM break points (%)	mean	std	ASE
2	-	-	3.0	0.374	0.040	0.307 (0.023)
3	3.5	0.187 (0.027)	89.0	0.238 0.548	0.072 0.089	0.211 (0.029)
4	6.5	0.157 (0.017)	8.0	0.156 0.391 0.667	0.045 0.062 0.093	0.182 (0.021)
5	15.5	0.170 (0.028)				
6	17.0	0.163 (0.025)				
7	20.0	0.158 (0.030)				
8	15.0	0.180 (0.029)				
9	11.5	0.203 (0.032)				
≥ 10	11.0	0.223 (0.035)				
All	100.0	0.18 (0.036)				0.211 (0.034)

estimates were obtained. The number of segments, locations of the break points and the ASEs of the Auto-PARM estimates are summarized in Table 11.

From this Table we observe that for most of the realizations Auto-PARM produces three segments. We computed the average of all the time-varying spectra of the 200 Auto-PARM estimates, the averaged spectrum is displayed in the right panel of Figure 9 and the average of the 200 Auto-SLEX estimates of the time-varying spectra is shown in the right panel of this figure.

The true spectrum in Figure 9 is well estimated by Auto-PARM and Auto-SLEX. Remarkably, Auto-PARM estimates well the true spectrum, in spite of the fact that it splits the realizations in fewer pieces than Auto-SLEX.

In Table 12 we summarize the Auto-PARM estimates of the AR orders for the above process

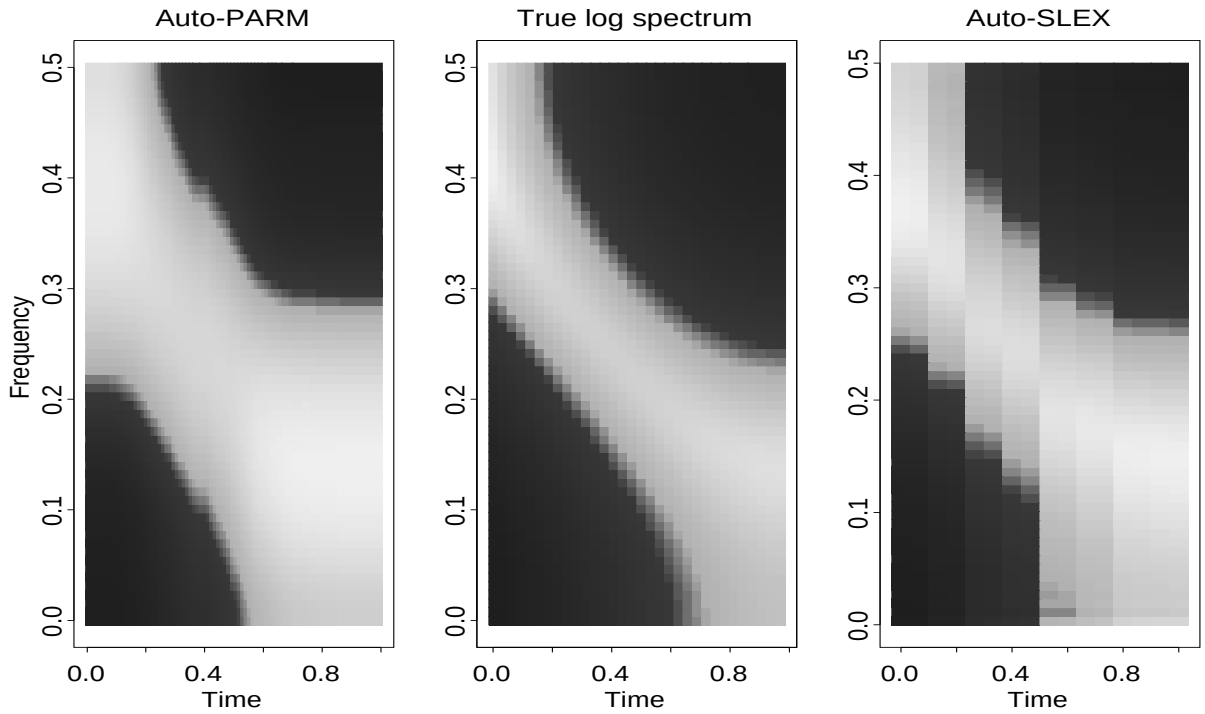

Figure 9: Left: True time-varying log-spectrum of process in (13) (center) and Auto-PARM and Auto-SLEX log-spectrum estimate (average of log-spectrum estimates obtained from 200 realizations.

for those realizations with 3 pieces. In general, the segments were modeled as AR processes of high order.

Table 12: Relative frequencies of the AR order selected by Auto-PARM for the realizations (with 3 segments) from the process (13).

Order	1	2	3	4	5
p_1	10.0	40.0	20.0	20.0	
p_2		40.0	20.0	30.0	
p_2		10.0	10.0	70.0	10.0

4.5 Short segments

To complement the above simulation experiments, we assess in this subsection the performance of Auto-PARM with the following process containing a short segment,

$$Y_t = \begin{cases} 0.75Y_{t-1} + \varepsilon_t, & \text{if } 1 \le t \le 50, \\ -0.50Y_{t-1} + \varepsilon_t, & \text{if } 51 \le t \le 1024, \end{cases} \tag{14}$$

where $\varepsilon_t \sim$ iid $\mathrm{N}(0, 1)$. A typical realization of this process is shown in Figure 10. For the

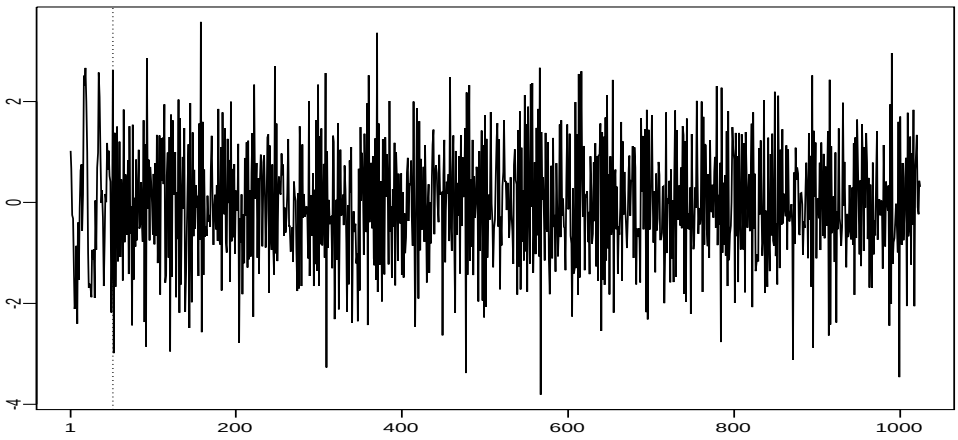

Figure 10: A realization from the piecewise stationary process in (14).

realization in Figure 10, Auto-PARM gives a single break point at $\hat{\tau}_1 = 51$ which is shown as the vertical dotted line in Figure 10. Both pieces are modeled as AR(1) processes. The run time for this realization was 2.70 seconds.

The Auto-PARM procedure was further applied to 200 realizations of this process. For all of these realizations Auto-PARM found one break point. The mean of the relative position estimates of this change point is 0.042 (true value is 0.049) with a standard error of 0.004. The minimum, median and maximum of the break points are 34, 51 and 70, respectively. In Table 13, the relative frequency of the orders p_1 and p_2 of each of the two pieces selected by Auto-PARM are shown. The Auto-PARM procedure segmented correctly 92.5% of the realizations (2 AR pieces of orders 1). This is exceptional performance for a process in which the break occurs near the beginning of the series.

Table 13: Relative frequencies of the AR order selected by Auto-PARM for the realizations from the process (14).

Order	0	1	2	3	≥ 4
p_1	0.0	96.0	3.0	0.5	0.0
p_2	0.0	96.0	4.0	0.0	0.0

Table 14: Summary of parameter estimates of the realizations of the process in (14) segmented correctly by Auto-PARM (92.5%) as two pieces, where each piece is an AR(1) process. For each segment, the true parameters, their mean and standard deviation (in parenthesis) are shown.

	1st piece		2nd piece	
parameter	ϕ_1	σ^2	ϕ_1	σ^2
true	0.75	1.00	-0.50	1.00
mean	0.66	1.05	-0.50	1.00
std	(0.11)	(0.23)	(0.03)	(0.04)

4.6 Further remarks on estimated breaks

As seen in the simulations from Sections 4.1 and 4.5, when the true unknown pieces are indeed AR processes, Auto-PARM can detect changes in order and in parameters. Let us consider for example the process in Section 4.1 where the first piece is an AR process of order 1 and the second piece is of order 2. In this case, Auto-PARM detected the change of order reasonably well (see Table 3). On the other hand, the second and third pieces of this process have the same order 2 with different parameter values. Also, the two pieces of the process in Section 4.5 have also the same order 1. Tables 3 and 13 show the Auto-PARM does a good job in detecting change of parameter values. The parameter estimates of both processes, given in Tables 4 and 8 respectively, show how well Auto-PARM also performs for parameter estimation.

The simulation in Section 4.3 is an example of a processes that is not a piece-wise AR processes. In this case, the first piece is an ARMA(1,1) process and the third piece is a MA(1) process. Auto-PARM approximates both the ARMA and MA pieces with AR processes perhaps of a large order. The fact that it did exceptionally well in detecting the breaks of this process (see Table 9) is not surprising, since for general stationary process, its spectral density can be well approximated by the spectrum of an AR process under the assumption of continuity of the spectral density (see, for example, Theorem 4.4.3, Brockwell and Davis 1991). The Auto-PARM procedure can then be interpreted as a method for segmenting piecewise stationary processes. In this example, the breaks Auto-PARM found are points where the spectrum has "large" changes.

5 APPLICATIONS

5.1 Seat Belt Legislation

In the hope of reducing the mean number of monthly "deaths and serious injuries", seat-belt legislation was introduced in UK on February 1983. Displayed in the left panel of Figure 11 is a time series $\{y_t\}_{t=1}^{120}$, beginning in January 1975, showing the monthly number of deaths and serious injuries. In order to remove the seasonal component of $\{y_t\}$, Brockwell and Davis (2002) consider the differenced time series $x_t = y_t - y_{t-12}$, and analyze $\{x_t\}$ with a regression model with errors following an ARMA model. The Auto-PARM procedure, when applied to the differenced series

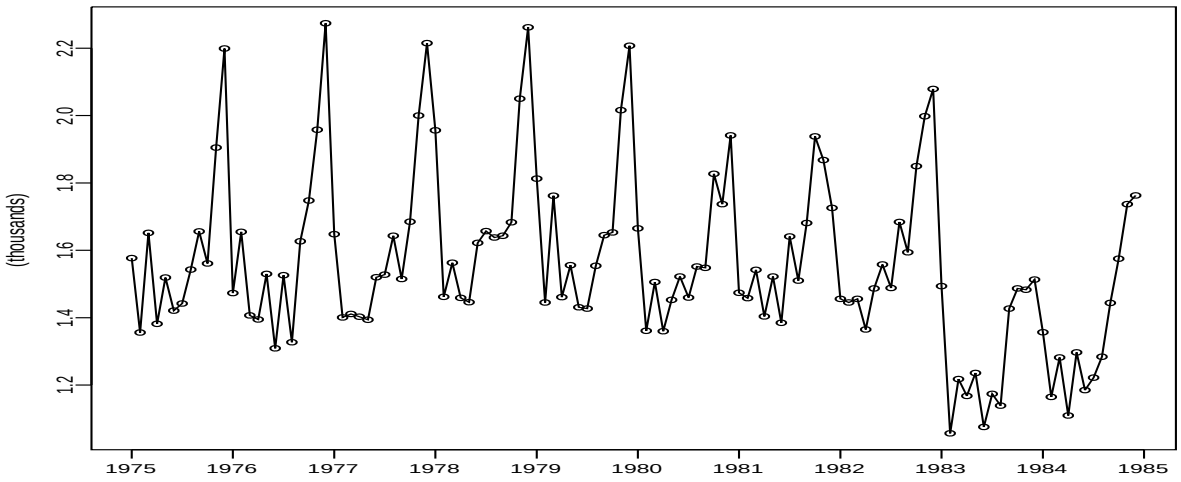

Figure 11: (a) Left panel. Monthly deaths and serious injuries on UK roads. (b) Right panel. Transformed seat belt legislation time series. The vertical lines are $\hat{\tau}_1$ and $\hat{\tau}_2$, respectively. The dotted horizontal line is the estimated mean of the i-th segment.

$\{x_t\}$, segmented the series into three pieces with break points at $\hat{\tau}_1 = 86$ and $\hat{\tau}_2 = 98$. The first two pieces are iid and the last piece is an AR process of order 1. On the right panel of Figure 11 the differenced time series $\{x_t\}$, along with the estimated means of each piece, are shown. From the Auto-PARM fit one can conclude that there is a structural change in the time series $\{y_t\}$ after February 1983, which coincides with the time of introduction of the seat belt legislation.

5.2 Speech Signal

The Auto-PARM procedure was applied to analyze a human speech signal which is the recording of the word "greasy". This signal contains 5762 observations and is shown at the top panel of Figure 12. This non-stationary time series was also analyzed by the Auto-SLEX procedure of Ombao et al. (2001). The Auto-PARM fit of this speech signal resulted in 15 segments. The total

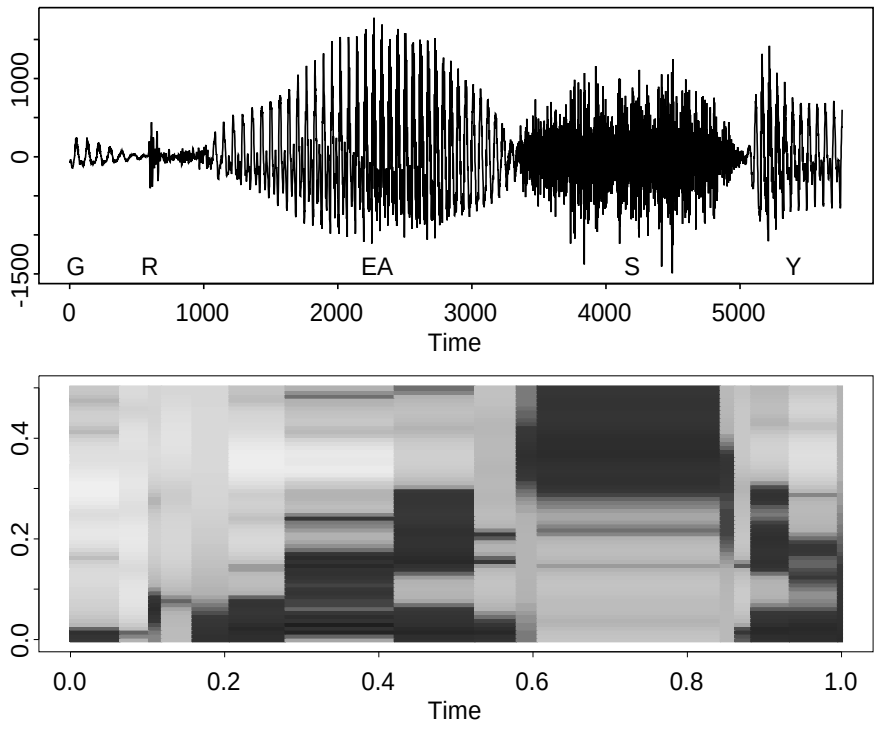

Figure 12: Top panel: Speech signal. Bottom panel: GA estimate of the time-varying log spectrum.

run time was 18.02 seconds. The time-varying log spectrum obtained with this fit is shown at the bottom panel of Figure 12. From this figure, one can see that the signal is roughly divided in segments that correspond to "G", "R", "EA", "S", and "Y". The information conveyed in this figure closely matches that from Ombao et al. (2001). The spectrum from those pieces that correspond to "G" have high power at the lowest frequencies. The pieces that correspond to "R" show power at frequencies slightly above that for "G". The pieces that correspond to "EA" show the evolution of power from lower to higher frequencies. The pieces that correspond to "S"

have high power at high frequencies. Notice that the Auto-PARM procedure breaks this speech signal into a smaller number of pieces than the Auto-SLEX procedure while still capturing the important features in the spectrum.

6 MULTIVARIATE TIME SERIES

In this section we demonstrate how Auto-PARM can be extended to model multivariate time series. In Subsection 6.1 the MDL of a piecewise multivariate autoregressive process is obtained and in Subsection 6.2 Auto-PARM is exemplified to a bivariate time series.

6.1 MDL

Let $\{\mathbf{Y}_t\}$ be a multivariate time series with r components, and assume that there are break points $\tau_0 := 1 < \tau_1 < \ldots < \tau_m < n+1$ for which the j-th piece $\mathbf{Y}_t = \mathbf{X}_{t,j}$, $\tau_{j-1} \leq t < \tau_j$, $j = 1, 2, \ldots, m+1$ is modeled by a multivariate AR(p_j) process

$$\mathbf{X}_{t,j} = \boldsymbol{\gamma}_j + \Phi_{j1}\mathbf{X}_{t-1,j} + \ldots + \Phi_{j,p_j}\mathbf{X}_{t-p_j,j} + \boldsymbol{\Sigma}_j^{1/2}\mathbf{Z}_t, \quad \tau_{j-1} \leq t < \tau_j, \tag{15}$$

where the noise sequence $\{\mathbf{Z}_t\}$ is iid with mean $\mathbf{0}$ and covariance matrix $\mathbf{I}$. The (unknown) AR matrix coefficients and covariance matrices are of dimension $r \times r$. Let $\mathcal{M}$ be the set of possible solutions for all the possible values of $m, \tau_1, \ldots, \tau_m, p_1, \ldots, p_m$. Let $\mathbf{y}_1, \ldots, \mathbf{y}_n$ be a realization of $\{\mathbf{Y}_t\}$. Parameter estimates in model (15) can be obtained using Whittle's algorithm (e.g., see Brockwell and Davis 1991). From (6), we have

$$\begin{aligned}\mathrm{MDL}(m, \tau_1, \ldots, \tau_m, p_1, \ldots, p_{m+1}) = \log m + (m+1)\log n + \sum_{j=1}^{m+1} \log p_j + \\ \sum_{j=1}^{m+1} \frac{3r + 2p_j r^2 + r^2}{4} \log n_j - \sum_{j=1}^{m+1} \log L(\hat{\Phi}_{j,1}, \ldots, \hat{\Phi}_{j,p_j}, \hat{\boldsymbol{\Sigma}}),\end{aligned}$$

where $L(\hat{\Phi}_{j,1}, \ldots, \hat{\Phi}_{j,p_j}, \hat{\boldsymbol{\Sigma}})$ is the likelihood of the j-th piece evaluated at the parameter estimates. As in the univariate case, the best segmentation of the realization $\mathbf{y}_1, \ldots, \mathbf{y}_n$ of $\{\mathbf{Y}_t\}$ is defined

as the minimizer of $\text{MDL}(m, \tau_1, \ldots, \tau_m, p_1, \ldots, p_{m+1})$. A similar GA can be developed for the practical minimization of $\text{MDL}(m, \tau_1, \ldots, \tau_m, p_1, \ldots, p_{m+1})$.

6.2 EEG analysis

Figure 13 displays two electroencephalograms (EEGs) each of length $n = 32768$ recorded from a female patient who was diagnosed with left temporal lobe epilepsy. This data set is courtesy of Dr. Beth Malow (formerly from the Department of Neurology at the University of Michigan). The top panel is the EEG from the left temporal lobe (T3 channel) while the bottom panel is the

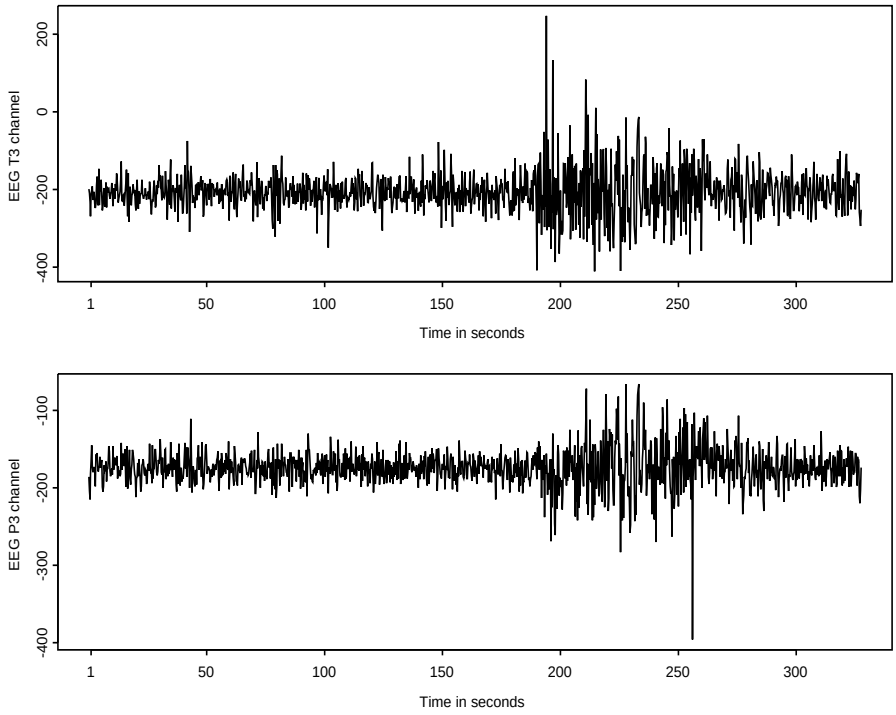

Figure 13: (*Bivariate EEG*) Electroencephalograms of length $n = 32768$ at channels T3 (top) and P3 (bottom) from a patient diagnosed with left temporal lobe epilepsy (courtesy of Dr. Beth Malow, formerly from the Deptarment of Neurology at the University of Michigan).

EEG from the left parietal lobe (P3 channel). Each EEG was recorded for a total of 5 minutes and 28 seconds with a sampling rate of 100 Hz. Of primary interest is the estimation of the power spectra of both EEGs and the coherence between them. One way of solving this problem is by segmenting the time series into stationary AR pieces (e.g., Gersch 1970; Jansen, Hasman,

Lenten, and Visser 1979; Ombao et al. 2001; Melkonian, Blumenthal and Meares 2003). The multivariate Auto-PARM procedure described above was applied to this bivariate time series, and the break point locations and the AR orders of the resulting fit are shown in Table 15. Notice that the multivariate implementation of Auto-PARM estimated the starting time for seizure for this epileptic episode at $t = 185.8$ seconds, in extremely close agreement with the neurologist's estimate of 185 seconds. In Figure 14, the estimated spectrums for the channel T3 (top panel) and channel P3 (bottom panel) based on the Auto-PARM fit in Table 15 are displayed. The estimates are close to those obtained in Ombao et al. (2001) and similar conclusions can be drawn. For example, prior to seizure, power was concentrated at lower frequencies. During seizure, power was spread to all frequencies, while towards the end of seizure, the concentration of power slowly restored to lower frequencies.

Table 15: GA segmentation of the bivariate time series from Figure 13. $\hat{\tau}_j$ is given in seconds.

	j											
	0	1	2	3	4	5	6	7	8	9	10	11
$\hat{\tau}_j$	1	185.8	189.6	206.1	220.9	233.0	249.0	261.6	274.6	306.0	308.4	325.8
$\hat{p}_j$	17	14	5	8	7	3	3	4	10	4	1	1

In Figure 15, the Auto-PARM estimate of the coherence between the T3 and P3 time series channels is shown. Again, this estimate is close to the estimate obtained in Ombao et al. (2001).

7 CONCLUSIONS

In this paper we provided a procedure to analyze a non-stationary time series by breaking it in pieces that are modeled as autoregressive processes. The best segmentation is obtained by minimizing a MDL criterion of the set of possible solutions via the genetic algorithm (our procedure does not make any restrictive assumptions on this set). The order of the autoregressive process and the estimates of the parameters of this process is a byproduct of this procedure. As seen

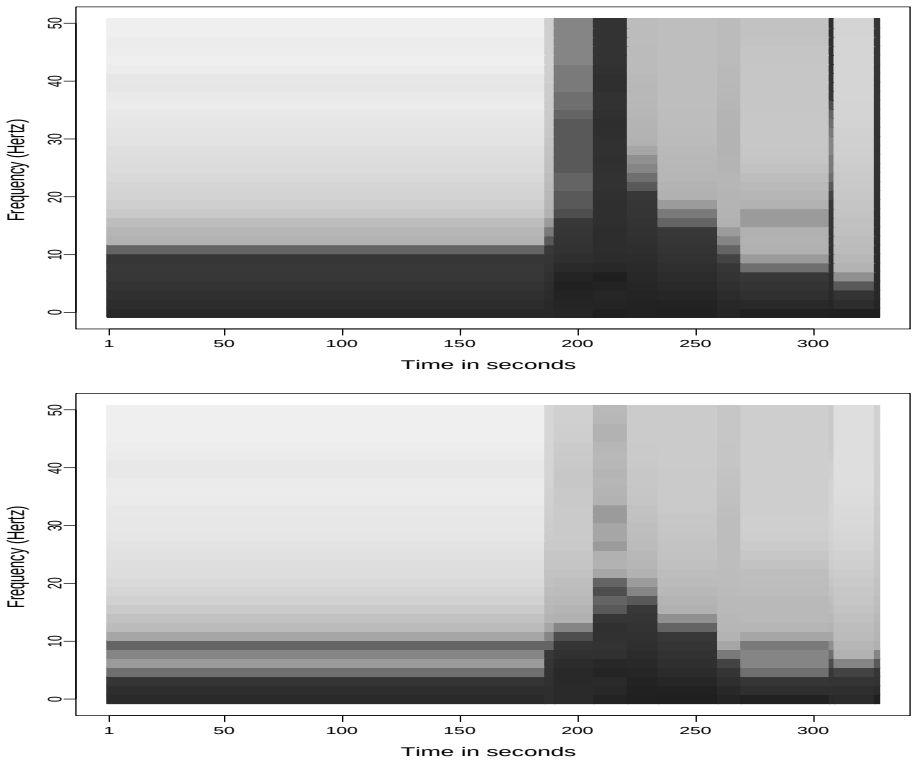

Figure 14: Estimate of the Time-varying log spectra of the EEGs from Figure 13. Top: T3 channel. Bottom: P3 channel.

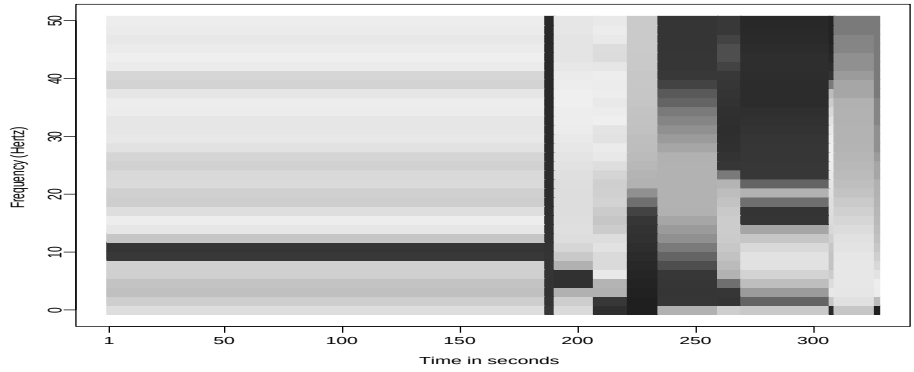

Figure 15: Estimated coherence between the EEGs shown in Figure 13.

in the simulation experiments, the rate in which this procedure segments correctly a piece-wise stationary process is high. Also, the "quality" of the estimated time-varying spectrum obtained with the results of our method is quite good.

APPENDIX: TECHNICAL DETAILS

In this Appendix we show the consistency of $\hat{\tau}_j/n, j = 1, \ldots, m$, when m, the number of breaks is known. Throughout this section we denote the true value of a parameter with a "0" superscript (except for σ_j^2). Preliminary results are given in Propositions A.2–A.4 and consistency is established in Proposition A.5.

Set $\lambda := (\lambda_1, \ldots, \lambda_m)$ and $p = (p_1, \ldots, p_{m+1})$. Since m is assumed known, for our asymptotic results notice that (6) can be rewritten in the compact form

$$2\text{MDL}(\lambda, p)/n = \frac{2(m+1)}{n}\log(n) + \sum_{j=1}^{m+1} \frac{p_j + 2}{n} \log n_j + \sum_{j=1}^{m+1} \frac{n_j}{n} \log(\hat{\sigma}_j^2) + o(1).$$

Proposition A.2 *Suppose $\{X_t\}$ is a stationary ergodic process with $E|X_t| < \infty$, then, with probability 1, the process*

$$S_n(s) = \frac{1}{n} \sum_{t=1}^{[ns]} X_t,$$

converges to the process sEX_1 on the space $D[0,1]$.

Proof. The argument relies on repeated application of the ergodic theorem. Let $\mathbb{Q}_{[0,1]}$ be the set of rational numbers in [0,1]. For $r \in \mathbb{Q}_{[0,1]}$,

$$\frac{1}{n} \sum_{t=1}^{[nr]} X_t \to rEX_1, \quad \text{a.s.} \tag{A.1}$$

If B_r is the set of ω's for which (A.1) holds, set

$$B = \bigcap_{r \in \mathbb{Q}_{[0,1]}} B_r,$$

and note $P(B) = 1$. Moreover, for $\omega \in B$ and any $s \in [0,1]$, choose $r_1, r_2 \in \mathbb{Q}_{[0,1]}$, such that $r_1 \leq s \leq r_2$. Hence,

$$|\frac{1}{n} \sum_{t=1}^{[ns]} X_t - \frac{1}{n} \sum_{t=1}^{[nr_1]} X_t| \leq \frac{1}{n} \sum_{t=[nr_1]}^{[nr_2]} |X_t| \to (r_2 - r_1)E|X_1|.$$

By making $|r_2 - r_1|$ arbitrarily small, it follows from the ergodic theorem that

$$\frac{1}{n}\sum_{t=1}^{[ns]} X_t \to sEX_1.$$

To establish convergence on $D[0,1]$, it suffices to show for $\omega \in B$

$$\frac{1}{n}\sum_{t=1}^{[ns]} X_t \to sEX_1, \quad \text{uniformly on [0,1].}$$

Given $\epsilon > 0$, choose $r_1, \ldots, r_m \in \mathbb{Q}_{[0,1]}$ such that $0 = r_0 < r_1 < \cdots < r_m = 1$, with $r_i - r_{i-1} < \epsilon$.

Then for any $s \in [0,1]$, $r_{i-1} < s \leq r_i$ and

$$|\frac{1}{n}\sum_{t=1}^{[ns]} X_t - sEX_1| \leq |\frac{1}{n}\sum_{t=1}^{[ns]} X_t - \frac{1}{n}\sum_{t=1}^{[nr_{i-1}]} X_t| + |\frac{1}{n}\sum_{t=1}^{[nr_{i-1}]} X_t - r_{i-1}EX_1| + |r_{i-1}EX_1 - sEX_1|.$$

The first term is bounded by

$$\frac{1}{n}\sum_{t=[nr_{i-1}]}^{[nr_i]} |X_t| \to (r_i - r_{i-1})E|X_1| < \epsilon E|X_1|.$$

Choose n so large that this term is less than $\epsilon E|X_1|$ for $i = 1, \ldots, m$. It follows that

$$\sup_s |\frac{1}{n}\sum_{t=1}^{[ns]} X_t - sEX_1| < \epsilon E|X_1| + \epsilon + \epsilon E|X_1|,$$

for n large. □

Proposition A.3 *Suppose $\{X_t\}$ is the AR(p_0) process*

$$X_t = \phi_0 + \phi_1 X_{t-1} + \ldots + \phi_{t-p_0} X_{t-p_0} + \sigma\varepsilon_t, \qquad \varepsilon_t \sim \textit{IID N(0,1).}$$

For $r, s \in$ [0,1] ($r < s$) and p=0,1,... P_0, let $\hat{\phi}(r,s,p)$ be the Y-W estimate of the AR(p) parameter vector $\phi(p)$ based on fitting an AR(p) to the data $X_{[rn]+1}, \ldots, X_{[sn]}$. Then with probability 1,

$$\hat{\phi}(r,s,p) \to \phi(p), \quad \hat{\sigma}^2(r,s,p) \to \sigma^2(p).$$

Proof. Since $\{X_t\}$ is a stationary ergodic process, $\{|X_t|\}$, $\{X_{t-i}X_{t-j}\}$ and $\{|X_{t-i}X_{t-j}|\}$ are stationary ergodic processes. By Proposition A.2, the partial sum processes for each of these processes converge to their respective limit a.s., let B be the probability 1 set on which these partial sum processes converge. Now $\hat{\phi}(r,s,p)$ and $\hat{\sigma}^2(r,s,p)$ are continuous functions of these processes. The result follows. □

Proposition A.4 *Let $\{Y_t\}$ be the process defined in (1) with $\phi_{0j} = 0$. For r,s ∈ [0,1] ($r < s$) and p=0,1,... P_0, let $\hat{\phi}_Y(r,s,p)$ be the Y-W estimates in fitting an AR(p) model to $Y_{[rn]+1}, \ldots, Y_{[sn]}$. Then with probability 1,*

$$\hat{\phi}_Y(r,s,p) \to \phi_Y^*(r,s,p), \quad \hat{\sigma}_Y^2(r,s,p) \to \sigma_Y^{*2}(r,s,p),$$

where $\phi_Y^*(r,s,p)$ and $\sigma_Y^{*2}(r,s,p)$ are defined below in the proof.

Proof. Let B_k^* be the probability 1 set on which

$$\frac{1}{n}\sum_{t=1}^{[ns]} X_{t,k}, \ \ \frac{1}{n}\sum_{t=1}^{[ns]} |X_{t,k}|, \ \frac{1}{n}\sum_{t=1}^{[ns]} X_{t-i,k}X_{t-j,k}, \ \text{ and } \ \frac{1}{n}\sum_{t=1}^{[ns]} |X_{t-i,k}X_{t-j,k}|, \ (i,j = 1,\ldots,P_0),$$

converge, $k = 1, 2, \ldots, m+1$, and set

$$B^* = \bigcap_{k=1}^{m+1} B_k^*.$$

Let $r, s \in [0,1]$, $r < s$, then $r \in [\lambda_{i-1}^0, \lambda_i^0)$ and $s \in (\lambda_{i-1+k}^0, \lambda_{i+k}^0]$, $k \geq 0$. Assuming that the mean of the process $\{Y_t\}$ is zero, we have

$$\begin{aligned}
\hat{\gamma}_Y(h) &:= \frac{1}{[sn]-[rn]} \sum_{t=[rn]+1}^{[sn]-h} Y_{t+h}Y_t \\
&= \frac{n}{[sn]-[rn]} \left\{ \frac{1}{n} \sum_{t=[rn]+1}^{[\lambda_i^0 n]-h} X_{t+h,i}X_{t,i} + \frac{1}{n} \sum_{t=[\lambda_i^0 n]+1}^{[\lambda_{i+1}^0 n]-h} X_{t+h,i+1}X_{t,i+1} \right. \\
&\qquad \left. + \cdots + \frac{1}{n} \sum_{t=[\lambda_{i-1+k}^0 n]+1}^{[sn]-h} X_{t+h,i+k}X_{t,i+k} + o(1) \right\}.
\end{aligned}$$

Let $\gamma_i(h) := \text{cov}\{X_{t+h,i}, X_{t,i}\}$. For $\omega \in B^*$, if follows from Proposition A.3 that

$$\begin{aligned}
\hat{\gamma}_Y(h) &\to \frac{\lambda_i^0 - r}{s-r}\gamma_i(h) + \frac{\lambda_{i+1}^0 - \lambda_i}{s-r}\gamma_{i+1}(h) + \cdots + \frac{s - \lambda_{i-1+k}^0}{s-r}\gamma_{i+k}(h), \\
&= a_i\gamma_i(h) + \ldots + a_{i+k}\gamma_{i+k}(h).
\end{aligned}$$

Then

$$\hat{\phi}_Y(r,s,p) = \hat{\mathbf{\Gamma}}_Y^{-1}(p)\hat{\boldsymbol{\gamma}}_Y(p) \to \left(\sum_{j=i}^{i+k} a_j \mathbf{\Gamma}_j(p)\right)^{-1} \sum_{j=i}^{i+k} a_j \boldsymbol{\gamma}_j(p) =: \phi_Y^*(r,s,p),$$

where $\mathbf{\Gamma}_j(p) = \{\gamma_j(i_1 - i_2)\}_{i_1,i_2=1}^p$ and $\boldsymbol{\gamma}_j(p) = [\gamma_j(1), \ldots, \gamma_j(p)]^T$. This establishes the desired convergence for $\hat{\phi}_Y(r,s,p)$. Note that if $k = 0$, $\phi_Y^*(r,s,p) = \phi_i(p)$. The proof of the convergence for $\hat{\sigma}_Y^2(r,s,p)$ is similar. □

Proposition A.5 *For the piecewise process in (1), choose $\epsilon > 0$ small such that*

$$\epsilon \ll \min_{i=1,\ldots,m+1} (\lambda_i^0 - \lambda_{i-1}^0)$$

and set

$$A_\epsilon = \{\lambda \in [0,1]^m, 0 = \lambda_0 < \lambda_1 < \lambda_2 < \ldots < \lambda_m < \lambda_{m+1} = 1,$$

$$\lambda_i - \lambda_{i-1} \geq \epsilon, i = 1, 2, \ldots, m+1\},$$

where $m = m^0$. If

$$\hat{\lambda}, \hat{p} = \arg \min_{\substack{\lambda \in A_\epsilon \\ 0 \leq p \leq P_0}} \frac{2}{n} \mathrm{MDL}(\lambda, p),$$

then $\hat{\lambda} \to \lambda^0$ a.s.

Proof Let B^* be the event described in the proof of Proposition A.4. We will show that for each $\omega \in B^*$, $\hat{\lambda} \to \lambda^0$. For $\omega \in B^*$, suppose $\hat{\lambda} \not\to \lambda^0$. Since the sequences are bounded, there exist a subsequence $\{n'_k\}$ such that $\hat{\lambda} \to \lambda^*$ and $\hat{p}_j \to p_j^*$ on the subsequence. Note that $\lambda^* \in A_\epsilon$ since $\hat{\lambda} \in A_\epsilon$ for all n. It follows that

$$\frac{2}{n} \mathrm{MDL}(\hat{\lambda}, \hat{p}) \to \sum_{j=1}^{m+1} (\lambda_j^* - \lambda_{j-1}^*) \log \sigma_Y^{*2}(\lambda_{j-1}^*, \lambda_j^*, p_j^*).$$

If $\lambda_i^0 \leq \lambda_{j-1}^* < \lambda_j^* \leq \lambda_{i+1}^0$, then

$$\sigma_Y^{*2}(\lambda_{j-1}^*, \lambda_j^*, p_j^*) = \sigma_{i+1}^2(p_j^*) \geq \sigma_{i+1}^2, \tag{A.9}$$

with equality if and only if $p_j^* \geq p_{i+1}$. If $\lambda_{i-1}^0 \leq \lambda_{j-1}^* < \lambda_i^0 < \cdots < \lambda_{i+k}^0 < \lambda_j^* \leq \lambda_{i+k+1}^0$, then

$$\sigma_Y^{*2}(\lambda_{j-1}^*, \lambda_j^*, p_j^*) \geq \frac{\lambda_i^0 - \lambda_{j-1}^*}{\lambda_j^* - \lambda_{j-1}^*} \sigma_i^2 + \frac{\lambda_{i+1}^0 - \lambda_i^0}{\lambda_j^* - \lambda_{j-1}^*} \sigma_{i+1}^2 + \cdots + \frac{\lambda_j^* - \lambda_{i+k}^0}{\lambda_j^* - \lambda_{j-1}^*} \sigma_{i+k+1}^2.$$

By the concavity of the log function,

$$\begin{aligned}(\lambda_j^* - \lambda_{j-1}^*) \log \sigma_Y^{*2}(\lambda_{j-1}^*, \lambda_j^*, p_j^*) &\geq (\lambda_j^* - \lambda_{j-1}^*) \left[\frac{\lambda_i^0 - \lambda_{j-1}^*}{\lambda_j^* - \lambda_{j-1}^*} \log \sigma_i^2 + \frac{\lambda_{i+1}^0 - \lambda_i^0}{\lambda_j^* - \lambda_{j-1}^*} \log \sigma_{i+1}^2 \right. \\ &\qquad \left. + \cdots + \frac{\lambda_j^* - \lambda_{i+k}^0}{\lambda_j^* - \lambda_{j-1}^*} \log \sigma_{i+k+1}^2 \right] \\ &= (\lambda_i^0 - \lambda_{j-1}^*) \log \sigma_i^2 + (\lambda_{i+1}^0 - \lambda_i^0) \log \sigma_{i+1}^2 \\ &\quad + \cdots + (\lambda_j^* - \lambda_{i+k}^0) \log \sigma_{i+k+1}^2.\end{aligned}$$

It follows that

$$\lim_{n\to\infty}\frac{2}{n}\mathrm{MDL}(\hat{\lambda},\hat{p}) > \sum_{i=1}^{m+1}(\lambda_i^0-\lambda_{i-1}^0)\log\sigma_i^2 = \lim_{n\to\infty}\frac{2}{n}\mathrm{MDL}(\lambda^0,p^0) \geq \lim_{n\to\infty}\frac{2}{n}\mathrm{MDL}(\hat{\lambda},\hat{p}), \quad \text{(A.10)}$$

a contradiction. Hence $\hat{\lambda}\to\lambda$ for all $\omega\in B^*$. $\square$

Notice that with probability 1, $\hat{p}_j$ can not underestimate p_j^0. To see this, let p_j^* as in the proof of Proposition (A.5), if for some j, $p_j^* < p_j^0$, then the contradiction in (A.10) is obtained again due to (A.9).

REFERENCES

Adak, S. (1998), "Time-dependent Spectral Analysis of Nonstationary Time Series," *Journal of the American Statistical Association*, 93, 1488-1501.

Alba, E., and Troya, J. M. (1999), "A Survey of Parallel Distributed Genetic Algorithm," *Complexity*, 4,31-52.

——- (2002), "Improving Flexibility and Efficiency by Adding Parallelism to Genetic Algorithms," *Statistics and Computing*, 12, 91-114.

Bai, J., and Perron, P. (1998), "Estimating and Testing Linear Models with Multiple Structural Changes," *Econometrica*, 66, 47-78.

——- (2003), "Computation and Analysis of Multiple Structural Change Models," *Journal of Applied Econometrics*, 18, 1-22.

Brockwell, P. J., and Davis, R. A. (1991), *Time Series: Theory and Methods*, (2nd ed.), New York: Springer-Verlag.

——- (2002), *Introduction to Time Series and Forecasting* (2nd ed.), New York: Springer-Verlag.

Chen, J., and Gupta, A. K. (1997), "Testing and Locating Variance Changepoints with Application to Stock Prices," *Journal of the American Statistical Association*, 92, 739-747.

Davis, L. D. (1991), *Handbook of Genetic Algorithms*, New York: Van Nostrand Reinhold.

Dahlhaus, R. (1997), "Fitting Time Series Models to Nonstationary Processes," *The Annals of Statistics*, 25, 1-37.

Forrest, S. (1991), *Emergent Computation*, Cambridge, MA: MIT Press.

Gaetan, C. (2000), "Subset ARMA Model Identification Using Genetic Algorithms", *Journal of Time Series Analysis*, 21, 5, 559-570.

Gersch, W. (1970), "Spectral Analysis of EEG's by Autoregressive Decomposition of Time Series," *Mathematical Biosciences*, 7, 205-222.

Hansen, M. H., and Yu, B. (2000), "Wavelet Thresholding via MDL for Natural Images", *IEEE Transactions on Information Theory*, 46, 1778-1788.

——- (2001), "Model Selection and the Principle of Minimum Description Length," *Journal of the American Statistical Association*, 96, 746-774.

Holland, J. (1975), *Adaptation in Natural and Artificial Systems*, Ann Arbor, MI: University of Michigan Press.

Inclan, C., and Tiao, G. C. (1994), "Use of Cumulative Sums of Squares for Retrospective Detection of Changes of Variance," *Journal of the American Statistical Association*, 89, 913-923.

Jansen, B. H., Hasman, A., Lenten, R., and Visser, S. L. (1979), "Usefulness of Autoregressive Models to Classify EEG-segments," *Biomedizinische Technik*, 24, 216-223.

Jornsten, R., and Yu, B. (2003), "Simultaneous Gene Clustering and Subset Selection for Classification via MDL," *Bioinformatics*, 19, 1100-1109.

Kim, C-J., and Nelson, C. R. (1999), *State-Space Models with Regime Switching*, Boston: MIT Press.

Kitagawa, G., and Akaike, H. (1978), "A Procedure for the Modeling of Non-stationary Time

Series," *Annals of the Institute of Statistical Mathematics*, 30, 351-363.

Lavielle, M. (1998), "Optimal Segmentation of Random Processes," *IEEE Transactions on Signal Processing*, 46, 1365-1373.

Lee, C-B. (1997), "Estimating the Number of Change Points in Exponential Families Distributions", *Scandinavian Journal of Statistics*, 24, 201-210.

Lee, Thomas C. M. (2000), "A Minimum Description Length Based Image Segmentation Procedure, and its Comparison with a Cross-Validation Based Segmentation Procedure", *Journal of the American Statistical Association*, 95, 259-270.

Lee, Thomas C. M., and Wong, Tan F. (2003), "Nonparametric Log-spectrum Estimation using Disconnected Regression Splines and Genetic Algorithms", *Signal Processing*, 83, 79-90.

Martin, W. N., Lienig, J., and Cohoon, J. P. (2000), "Island (Migration) Models: Evolutionary Algorithm Based on Punctuated Equilibria," in *Evolutionary Computation (Vol. 2), Advanced Algorithms and Operators*, ed. D.B. Fogel, 101-124. Philadelphia: Bristol.

Melkonian, D., Blumenthal, T. D., and Meares, R. (2003), "High-resolution Fragmentary Decomposition -a Model Based Method of Non-stationary Electrophysiological Signal Analysis," *Journal of Neuroscience Methods*, 131, 149-159.

Ombao, H. C., Raz, J. A. Von Sachs, R., and Malow, B. A. (2001), "Automatic Statistical Analysis of Bivariate Nonstationary Time Series," *Journal of the American Statistical Association*, 96, 543-560.

Pittman, J. (2002), "Adaptive Splines and Genetic Algorithms", *Journal of Computational and Graphical Statistics*, 11, 3, 1-24.

Punskaya, E., Andrieu, C., Doucet, A., and Fitzgerald, W. J. (2002), "Bayesian Curve Fitting Using MCMC With Applications to Signal Segmentation," *IEEE Transactions on Signal Pro-*

cessing, 50, 747-758.

Rissanen, J. (1989), *Stochastic Complexity in Statistical Inquiry*, Singapore: World Scientific.

Saito, N. (1994), "Simultaneous Noise Suppression and Signal Compression Using a Library of Ortho-normal Bases and the Minimum Description Length Criterion," in *Wavelets and Gheophysics*, eds. E Foufoula-Georgiou and P. Kumar, 299-324. New York: Academic Press.

Yao, Y-C. (1988), "Estimating the Number of Change-points via Schwarz' Criterion", *Statistics & Probability Letters*, 6, 181-189.