Structural Break Detection in Time Series

Richard A. Davis
Thomas Lee
Gabriel Rodriguez-Yam

Colorado State University
(http://www.stat.colostate.edu/~rdavis/lectures)

This research supported in part by an IBM faculty award.
Introduction
  • Background
  • Setup for AR models

Model Selection Using Minimum Description Length (MDL)
  • General principles
  • Application to AR models with breaks

Optimization using Genetic Algorithms
  • Basics
  • Implementation for structural break estimation

Simulation examples
  • Piecewise autoregressions
  • Slowly varying autoregressions

Applications
  • Multivariate time series
  • EEG example

Application to nonlinear models (parameter-driven state-space models)
  • Poisson model
  • Stochastic volatility model
Introduction

Structural breaks:

Kitagawa and Akaike (1978)
- fitting locally stationary autoregressive models using AIC
- computations facilitated by the use of the Householder transformation

Davis, Huang, and Yao (1995)
- likelihood ratio test for testing a change in the parameters and/or order of an AR process.

Kitagawa, Takanami, and Matsumoto (2001)
- signal extraction in seismology-estimate the arrival time of a seismic signal.

Ombao, Raz, von Sachs, and Malow (2001)
- orthogonal complex-valued transforms that are localized in time and frequency- smooth localized complex exponential (SLEX) transform.
- applications to EEG time series and speech data.
Introduction (cont)

Locally stationary:

Dahlhaus (1997, 2000, …)
  • locally stationary processes
  • estimation

Adak (1998)
  • piecewise stationary
  • applications to seismology and biomedical signal processing

MDL and coding theory:

  • estimation of discontinuous regression functions

Hansen and Yu (2001)
  • model selection
Time Series: \( y_1, \ldots, y_n \)

**Piecewise AR model:**

\[
Y_t = \gamma_j + \phi_{j1} Y_{t-1} + \cdots + \phi_{j p_j} Y_{t-p_j} + \sigma_j \varepsilon_t, \quad \text{if } \tau_{j-1} \leq t < \tau_j,
\]

where \( \tau_0 = 1 < \tau_1 < \cdots < \tau_{m-1} < \tau_m = n + 1 \), and \( \{\varepsilon_t\} \) is IID(0,1).

**Goal:** Estimate

- \( m \) = number of segments
- \( \tau_j \) = location of \( j \)th break point
- \( \gamma_j \) = level in \( j \)th epoch
- \( p_j \) = order of AR process in \( j \)th epoch
- \((\phi_{j1}, \ldots, \phi_{jp_j})\) = AR coefficients in \( j \)th epoch
- \( \sigma_j \) = scale in \( j \)th epoch
Motivation for using piecewise AR models:

Piecewise AR is a special case of a **piecewise stationary process** (see Adak 1998),

\[
\widetilde{Y}_{t,n} = \sum_{j=1}^{m} Y_t^j I_{[\tau_{j-1}, \tau_j)}(t / n),
\]

where \( \{Y_t^j\}, j = 1, \ldots, m \) is a sequence of stationary processes. It is argued in Ombao et al. (2001), that if \( \{Y_{t,n}\} \) is a locally stationary process (in the sense of Dahlhaus), then there exists a piecewise stationary process \( \{\widetilde{Y}_{t,n}\} \) with

\[
m_n \rightarrow \infty \quad \text{with} \quad m_n / n \rightarrow 0, \quad \text{as} \quad n \rightarrow \infty,
\]

that approximates \( \{Y_{t,n}\} \) (in average mean square).

Roughly speaking: \( \{Y_{t,n}\} \) is a locally stationary process if it has a time-varying spectrum that is approximately \( |A(t/n, \omega)|^2 \), where \( A(u, \omega) \) is a continuous function in \( u \).
Example--Monthly Deaths & Serious Injuries, UK

Data: \( y_t = \) number of monthly deaths and serious injuries in UK, Jan `75 – Dec `84, \((t = 1, \ldots, 120)\)

Remark: Seat belt legislation introduced in Feb `83 \((t = 99)\).
Example -- Monthly Deaths & Serious Injuries, UK (cont)

Data: \( x_t \) = number of monthly deaths and serious injuries in UK, differenced at lag 12; Jan `75 – Dec `84, \((t = 13, \ldots, 120)\)

Remark: Seat belt legislation introduced in Feb `83 \((t = 99)\).

Traditional regression analysis:

\[
Y_t = a + bf(t) + W_t,
\]

\[
f(t) = \begin{cases} 
0, & \text{if } 1 \leq t \leq 98, \\
1, & \text{if } 98 < t \leq 120. 
\end{cases}
\]

\[
X_t = Y_t - Y_{t-12}
\]

\[
= bg(t) + N_t
\]

\[
g(t) = \begin{cases} 
1, & \text{if } 99 \leq t \leq 110, \\
0, & \text{otherwise.}
\end{cases}
\]

Model: \( b = -373.4, \{N_t\} \sim AR(13)\).
Model Selection Using Minimum Description Length

Basics of MDL:
Choose the model which *maximizes the compression* of the data or, equivalently, select the model that *minimizes the code length* of the data (i.e., amount of memory required to encode the data).

\[ M = \text{class of operating models for } y = (y_1, \ldots, y_n) \]

\[ L_F(y) = \text{code length of } y \text{ relative to } F \in M \]

Typically, this term can be decomposed into two pieces (two-part code),

\[ L_F(y) = L(\hat{F}|y) + L(\hat{e}|\hat{F}) \]

where

\[ L(\hat{F}|y) = \text{code length of the fitted model for } F \]

\[ L(\hat{e}|\hat{F}) = \text{code length of the residuals based on the fitted model} \]
Encoding the data: \((x_1, y_1), \ldots, (x_n, y_n)\)

1. "Naïve" case

\[
L("naïve") = L(x_1, \ldots, x_n) + L(y_1, \ldots, y_n) \\
= L(x_1) + \cdots + L(x_n) + L(y_1) + \cdots + L(y_n)
\]

2. Linear model; suppose \(y_i = a_0 + a_1 x_i, \ i = 1, \ldots, n\). Then

\[
L("p = 1") = L(x_1, \ldots, x_n) + L(a_0, a_1) \\
= L(x_1) + \cdots + L(x_n) + L(a_0) + L(a_1)
\]

3. Linear model with noise; suppose \(y_i = a_0 + a_1 x_i + \varepsilon_i, \ i = 1, \ldots, n\), where \(\{\varepsilon_i\} \sim \text{IID } N(0, \sigma^2)\). Then

\[
L("p = 1") = L(x_1) + \cdots + L(x_n) + L(\hat{a}_0) + L(\hat{a}_1) + L(\hat{\sigma}^2) + L(\hat{\varepsilon}_1, \ldots, \hat{\varepsilon}_n | \hat{a}_0, \hat{a}_1, \hat{\sigma}^2) \\
A
\]

If \(A < L(y_1) + \ldots + L(y_n)\), then "p=1" encoding scheme dominates the "naïve" scheme.
Model Selection Using Minimum Description Length (cont)

Applied to the segmented AR model:

\[ Y_t = \gamma_j + \phi_{j1} Y_{t-1} + \cdots + \phi_{jp_j} Y_{t-p_j} + \sigma_j \varepsilon_t, \quad \text{if } \tau_{j-1} \leq t < \tau_j, \]

First term  \( L(\hat{F}_y) \): Let \( n_j = \tau_j - \tau_{j-1} \); \( \psi_j = (\gamma_j, \phi_{j1}, \ldots, \phi_{jp_j}, \sigma_j) \)
denote the length of the \( j \)th segment and the parameter vector of
the \( j \)th AR process, respectively. Then

\[
L(\hat{F}_y) = L(m) + L(\tau_1, \ldots, \tau_m) + L(p_1, \ldots, p_m) + L(\hat{\psi}_1 | y) + \cdots + L(\hat{\psi}_m | y)
= L(m) + L(n_1, \ldots, n_m) + L(p_1, \ldots, p_m) + L(\hat{\psi}_1 | y) + \cdots + L(\hat{\psi}_m | y)
\]

Encoding:

integer \( l \): \( \log_2 l \) bits (if \( l \) unbounded)
\[
\log_2 l_U \text{ bits (if } l \text{ bounded by } l_U) \]

MLE \( \hat{\theta} \): \( \frac{1}{2} \log_2 N \) bits (where \( N = \text{number of observations used to compute } \hat{\theta} \); Rissanen (1989))
So, 
\[ L(\hat{F}|y) = \log_2 m + m \log_2 n + \sum_{j=1}^{m} \log_2 p_j + \sum_{j=1}^{m} \frac{p_j + 2}{2} \log_2 n_j \]

Second term \( L(\hat{e} | \hat{F}) \) : Using Shannon’s classical results on information theory, Rissanen demonstrates that the code length of \( \hat{e} \) can be approximated by the negative of the log-likelihood of the fitted model, i.e., by
\[
L(\hat{e} | \hat{F}) \approx \sum_{j=1}^{m} \frac{n_j}{2} \left( \log_2 (2\pi \hat{\sigma}_j^2) + 1 \right)
\]

For fixed values of \( m, (\tau_1, p_1), \ldots, (\tau_m, p_m) \), we define the MDL as
\[
MDL(m, (\tau_1, p_1), \ldots, (\tau_m, p_m))
\]
\[
= \log_2 m + m \log_2 n + \sum_{j=1}^{m} \log_2 p_j + \sum_{j=1}^{m} \frac{p_j + 2}{2} \log_2 n_j + \sum_{j=1}^{m} \frac{n_j}{2} \log_2 (2\pi \hat{\sigma}_j^2) + \frac{n}{2}
\]

The strategy is to find the best segmentation that minimizes \( MDL(m, \tau_1, p_1, \ldots, \tau_m, p_m) \). To speed things up, we use Y-W estimates of AR parameters.
Optimization Using Genetic Algorithms

Basics of GA:
Class of optimization algorithms that mimic natural evolution.

- Start with an initial set of *chromosomes*, or population, of possible solutions to the optimization problem.
- Parent chromosomes are randomly selected (proportional to the rank of their objective function values), and produce offspring using *crossover* or *mutation* operations.
- After a sufficient number of offspring are produced to form a second generation, the process then *restarts to produce a third generation*.
- Based on Darwin’s *theory of natural selection*, the process should produce future generations that give a *smaller (or larger)* objective function.
Application to Structural Breaks—(cont)

Genetic Algorithm: Chromosome consists of \( n \) genes, each taking the value of \(-1\) (no break) or \( p \) (order of AR process). Use natural selection to find a *near* optimal solution.

Map the break points with a chromosome \( c \) via

\[
(m, (\tau_1, p_1), \ldots, (\tau_m, p_m)) \leftrightarrow c = (\delta_1, \ldots, \delta_n),
\]

where

\[
\delta_t = \begin{cases} 
-1, & \text{if no break point at } t, \\
p_j, & \text{if break point at time } t = \tau_{j-1} \text{ and AR order is } p_j.
\end{cases}
\]

For example,

\[
c = (2, -1, -1, -1, -1, 0, -1, -1, -1, 0, -1, -1, -1, 3, -1, -1, -1, -1, -1, -1, -1)
\]

\[
t: 1 \quad 6 \quad 11 \quad 15
\]

would correspond to a process as follows:

\[
\text{AR}(2), t=1:5; \text{AR}(0), t=6:10; \text{AR}(0), t=11:14; \text{AR}(3), t=15:20
\]
Implementation of Genetic Algorithm—(cont)

Generation 0: Start with $L$ (200) randomly generated chromosomes, $c_1, \ldots, c_L$ with associated MDL values, $MDL(c_1), \ldots, MDL(c_L)$.

Generation 1: A new child in the next generation is formed from the chromosomes $c_1, \ldots, c_L$ of the previous generation as follows:

- with probability $\pi_c$, crossover occurs.
  - two parent chromosomes $c_i$ and $c_j$ are selected at random with probabilities proportional to the ranks of $MDL(c_i)$.
  - $k^{th}$ gene of child is $\delta_k = \delta_{i,k}$ w.p. $\frac{1}{2}$ and $\delta_{j,k}$ w.p. $\frac{1}{2}$

- with probability $1 - \pi_c$, mutation occurs.
  - a parent chromosome $c_i$ is selected
  - $k^{th}$ gene of child is $\delta_k = \delta_{i,k}$ w.p. $\pi_1$; $-1$ w.p. $\pi_2$; and $\rho$ w.p. $1 - \pi_1 - \pi_2$. 
Execution of GA: Run GA until convergence or until a maximum number of generations has been reached.

Various Strategies:

- include the top ten chromosomes from last generation in next generation.
- use multiple islands, in which populations run independently, and then allow migration after a fixed number of generations. This implementation is amenable to parallel computing.
1. Piecewise stationary with dyadic structure: Consider a time series following the model,

\[
Y_t = \begin{cases} 
.9Y_{t-1} + \varepsilon_t, & \text{if } 1 \leq t < 513, \\
1.69Y_{t-1} - .81Y_{t-2} + \varepsilon_t, & \text{if } 513 \leq t < 769, \\
1.32Y_{t-1} - .81Y_{t-2} + \varepsilon_t, & \text{if } 769 \leq t \leq 1024,
\end{cases}
\]

where \(\{\varepsilon_t\} \sim \text{IID N}(0,1)\).
1. Piecewise stat (cont)

Implementation: Start with $NI = 50$ islands, each with population size $L = 200$.

After every $Mi = 5$ generations, allow migration.

Replace worst 2 in Island 2 with best 2 from Island 4.

Stopping rule: Stop when the max MDL does not change for 10 consecutive migrations or after 100 migrations.

Span configuration for model selection: Max AR order $K = 10$,

<table>
<thead>
<tr>
<th>$p$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7-10</th>
<th>11-20</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m_p$</td>
<td>10</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td>16</td>
<td>18</td>
<td>20</td>
<td>25</td>
<td>50</td>
</tr>
<tr>
<td>$\pi_p$</td>
<td>1/21</td>
<td>1/21</td>
<td>1/21</td>
<td>1/21</td>
<td>1/21</td>
<td>1/21</td>
<td>1/21</td>
<td>1/21</td>
<td>1/21</td>
</tr>
</tbody>
</table>
1. Piecewise stat (cont)

GA results: 3 pieces breaks at $\tau_1=513; \tau_2=769$. Total run time 16.31 secs

Fitted model:

<table>
<thead>
<tr>
<th></th>
<th>$\phi_1$</th>
<th>$\phi_2$</th>
<th>$\sigma^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1- 512:</td>
<td>0.857</td>
<td></td>
<td>0.9945</td>
</tr>
<tr>
<td>513- 768:</td>
<td>1.68</td>
<td>-0.801</td>
<td>1.1134</td>
</tr>
<tr>
<td>769-1024:</td>
<td>1.36</td>
<td>-0.801</td>
<td>1.1300</td>
</tr>
</tbody>
</table>

True Model

Fitted Model
1. Piecewise stat (cont)

Simulation: 200 replicates of time series of length 1024 were generated. (SLEX results from Ombao et al.)

\[
ASE = n^{-1}(1/33) \sum_{t=1}^{n} \sum_{j=0}^{32} \left\{ \log \hat{f}(t/n, \omega_j) - \log f(t/n, \omega_j) \right\}^2, \quad \omega_j = 2\pi j / 64.
\]

<table>
<thead>
<tr>
<th># of segments</th>
<th>%</th>
<th>Auto-SLEX</th>
<th>Change Points</th>
<th>ASE</th>
<th>%</th>
<th>mean</th>
<th>std</th>
<th>ASE</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0</td>
<td>1/2</td>
<td></td>
<td>-</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>60.0</td>
<td>1/4 , 3/4</td>
<td></td>
<td>25.42(4.56)</td>
<td>82.0</td>
<td>.500</td>
<td>.006</td>
<td>3.64 (.13)</td>
</tr>
<tr>
<td>4</td>
<td>34.0</td>
<td>1/4, 2/4, 3/4</td>
<td></td>
<td>34.09 (6.74)</td>
<td>17.5</td>
<td>.476</td>
<td>.080</td>
<td>3.73 (.13)</td>
</tr>
<tr>
<td>5</td>
<td>5.0</td>
<td>2/8, 4/8, 5/8, 6/8, 7/8</td>
<td></td>
<td>32.77 (5.20)</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\geq 6</td>
<td>1.0</td>
<td></td>
<td></td>
<td>50.01 (6.25)</td>
<td>0.5</td>
<td></td>
<td></td>
<td>3.83</td>
</tr>
</tbody>
</table>
1. Piecewise stat (cont)

Simulation (cont):

True model:

\[ Y_t = \begin{cases} 
.9Y_{t-1} + \varepsilon_t, & \text{if } 1 \leq t < 513, \\
1.69Y_{t-1} - .81Y_{t-2} + \varepsilon_t, & \text{if } 513 \leq t < 769, \\
1.32Y_{t-1} - .81Y_{t-2} + \varepsilon_t, & \text{if } 769 \leq t \leq 1024, 
\end{cases} \]

AR orders selected (percent):

<table>
<thead>
<tr>
<th>Order</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>≥ 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>(p_1)</td>
<td>0</td>
<td>99.4</td>
<td>0.60</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(p_2)</td>
<td>0</td>
<td>0</td>
<td>86.0</td>
<td>11.6</td>
<td>1.8</td>
<td>0.6</td>
<td>0</td>
</tr>
<tr>
<td>(p_3)</td>
<td>0</td>
<td>0</td>
<td>89.0</td>
<td>10.4</td>
<td>0.6</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Simulation Examples (cont)

3. Piecewise stationary with short segments:

\[
Y_t = \begin{cases} 
0.9Y_{t-1} + \varepsilon_t, & \text{if } 1 \leq t < 51 \\
0.25Y_{t-1} + \varepsilon_t, & \text{if } 50 \leq t \leq 500 
\end{cases}
\]

where \( \{\varepsilon_t\} \sim \text{IID } N(0,1) \).

GA results: 2 pieces with break at \( \tau_1 = 47 \)
Simulation results: Change occurred at time $\tau_1 = 51$; $51/500 = .1$

<table>
<thead>
<tr>
<th># of segments</th>
<th>change points</th>
<th>%</th>
<th>mean</th>
<th>std</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>9.0</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>89.0</td>
<td>.096</td>
<td>.017</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>2.0</td>
<td>.048</td>
<td>.020</td>
</tr>
<tr>
<td>≥ 4</td>
<td></td>
<td>0</td>
<td>.092</td>
<td>.011</td>
</tr>
</tbody>
</table>
Simulation Examples (cont)

4. Slowly varying AR(2) model:

\[ Y_t = a_t Y_{t-1} - 0.81 Y_{t-2} + \varepsilon_t \quad \text{if} \quad 1 \leq t \leq 1024 \]

where \( a_t = 0.8[1 - 0.5\cos(\pi t / 1024)] \), and \( \{\varepsilon_t\} \sim \text{IID } \text{N}(0,1) \).
4. Slowly varying AR(2) (cont)

GA results: 3 pieces, breaks at $\tau_1=293$, $\tau_2=615$. Total run time 27.45 secs

Fitted model:

<table>
<thead>
<tr>
<th></th>
<th>$\phi_1$</th>
<th>$\phi_2$</th>
<th>$\sigma^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-292</td>
<td>0.365</td>
<td>-0.753</td>
<td>1.149</td>
</tr>
<tr>
<td>293-614</td>
<td>0.821</td>
<td>-0.790</td>
<td>1.176</td>
</tr>
<tr>
<td>615-1024</td>
<td>1.084</td>
<td>-0.760</td>
<td>0.960</td>
</tr>
</tbody>
</table>
4. Slowly varying AR(2) (cont)

In the graph below right, we average the spectogram over the GA fitted models generated from each of the 200 simulated realizations.
Example: Monthly Deaths & Serious Injuries, UK

Data: $Y_t =$ number of monthly deaths and serious injuries in UK, Jan `75 – Dec `84, ($t = 1, \ldots, 120$)

Remark: Seat belt legislation introduced in Feb `83 ($t = 99$).
Example: Monthly Deaths & Serious Injuries, UK

Data: $Y_t =$ number of monthly deaths and serious injuries in UK, Jan `75 – Dec `84, ($t = 1, \ldots, 120$)

Remark: Seat belt legislation introduced in Feb `83 ($t = 99$).

Results from GA: 3 pieces; time = 4.4secs

Piece 1: ($t=1,\ldots, 98$) IID; Piece 2: ($t=99,\ldots,108$) IID; Piece 3: $t=109,\ldots,120$ AR(1)
Examples

Speech signal: GREASY
Speech signal: GREASY

\( n = 5762 \) observations

\( m = 15 \) break points

Run time = 18.02 secs
Examples

Large brown bat echolocation: 400 data points taken at 7 microsecond intervals (total duration of .0028 seconds). Data and ideas about M-stationarity described here are from Buddy Gray, Wayne Woodward, and their group at SMU. http://faculty.smu.edu/hgray/research.htm

bat echolocation
Features of data:

• *time varying frequency*, examples of which are *chirps* and *doppler signals* found in *radar*, *sonar*, and *communication theory*.

• data appears to be made up of *two signals*.

• each signal has a *frequency* that is changing linearly in time. i.e., that is the *cycle is lengthening* in time.

• an AR(20) model is the *best fitting* AR model. Residuals are *uncorrelated* but *not independent*.
Examples (bat data cont)

M-Stationarity (Gray et al): \( \text{Cov}(Y(t), Y(t\tau)) = R(\tau) \).

- This notion corresponds to a *time-deformation* (logarithmic in this case) to make the transformed process stationary in the ordinary sense.

- The *Euler* process (Gray and Zhang `98) is an example of an M-stationary process.
Examples (bat data cont)

GA results: 6 pieces, breaks at $\tau_1=16$, $\tau_2=98$, $\tau_3=205$, $\tau_4=265$, $\tau_5=353$.
Fitted model: AR orders 1, 6, 13, 7, 13, 5; Total run time 4.7 secs
Examples (bat data spectrograms)

Euler(12), Gray et al

Auto-PARM
Application to Multivariate Time Series

Multivariate time series (d-dimensional): $y_1, \ldots, y_n$

Piecewise AR model:

$$Y_t = \gamma_j + \Phi_{j1} Y_{t-1} + \cdots + \Phi_{jp_j} Y_{t-p_j} + \sum_{j=1}^{1/2} z_t, \quad \text{if } \tau_{j-1} \leq t < \tau_j,$$

where $\tau_0 = 1 < \tau_1 < \ldots < \tau_{m-1} < \tau_m = n + 1$, and $\{Z_t\}$ is IID($0$, $I_d$).

In this case,

$$MDL(m, (\tau_1, p_1), \ldots, (\tau_m, p_m)) = \log m + m \log n + \sum_{j=1}^{m} \log p_j$$

$$+ \sum_{j=1}^{m} \frac{p_j d^2 + d + d(d + 1)/2}{2} \log n_j + \sum_{j=1}^{m} \frac{1}{2} \sum_{t=\tau_{j-1}}^{\tau_j-1} \left( \log |\hat{V}_t| + (Y_t - \hat{Y}_t)^T \hat{V}_t^{-1} (Y_t - \hat{Y}_t) \right),$$

where $\hat{Y}_t = E(Y_t | Y_{t-1}, \ldots, Y_1)$ and $\hat{V}_t = E(Y_t - \hat{Y}_t)^2$ and the AR parameters are estimated by the multivariate Y-W equations based on Whittle’s generalization of the Durbin-Levinson algorithm.
Example: EEG Time series

Data: Bivariate EEG time series at channels T3 (left temporal) and P3 (left parietal). Female subject was diagnosed with left temporal lobe epilepsy. Data collected by Dr. Beth Malow and analyzed in Ombao et al (2001). (n=32,768; sampling rate of 100H). Seizure started at about 1.85 seconds.

GA bivariate results: 11 pieces with AR orders, 17, 2, 6 15, 2, 3, 5, 9, 5, 4, 1
GA univariate results: 14 breakpoints for T3; 11 breakpoints for P3

Data: Bivariate EEG time series at channels T3 (left temporal) and P3 (left parietal). Female subject was diagnosed with left temporal lobe epilepsy. Data collected by Dr. Beth Malow and analyzed in Ombao et al (2001). (n=32,768; sampling rate of 100H). Seizure started at about 1.85 seconds.

GA bivariate results: 11 pieces with AR orders, 17, 2, 6 15, 2, 3, 5, 9, 5, 4, 1
GA univariate results: 14 breakpoints for T3; 11 breakpoints for P3

T3 Channel

P3 Channel
Remarks:

- the general conclusions of this analysis are similar to those reached in Ombao et al.

- prior to seizure, power concentrated at lower frequencies and then spread to high frequencies.

- power returned to the lower frequencies at conclusion of seizure.
Remarks (cont):

- T3 and P3 strongly coherent at 9-12 Hz prior to seizure.
- Strong coherence at low frequencies just after onset of seizure.
- Strong coherence shifted to high frequencies during the seizure.

Example: EEG Time series (cont)
Application to Parameter-Driven SS Models

State Space Model Setup:

Observation equation:

\[ p(y_t | \alpha_t) = \exp\{\alpha_t y_t - b(\alpha_t) + c(y_t)\}. \]

State equation: \{\alpha_t\} follows the piecewise AR(1) model given by

\[ \alpha_t = \gamma_k + \phi_k \alpha_{t-1} + \sigma_k \varepsilon_t, \quad \text{if} \quad \tau_{k-1} \leq t < \tau_k, \]

where \( 1 = \tau_0 < \tau_1 < \ldots < \tau_m < n \), and \( \{\varepsilon_t\} \sim \text{IID N}(0,1) \).

Parameters:

\[ m = \text{number of break points} \]
\[ \tau_k = \text{location of break points} \]
\[ \gamma_k = \text{level in } k^{\text{th}} \text{ epoch} \]
\[ \phi_k = \text{AR coefficients } k^{\text{th}} \text{ epoch} \]
\[ \sigma_k = \text{scale in } k^{\text{th}} \text{ epoch} \]
Estimation: For \((m, \tau_1, \ldots, \tau_m)\) fixed, calculate the approximate likelihood evaluated at the “MLE”, i.e.,

\[
L_a(\hat{\psi}; y_n) = \frac{|G_n|^{1/2}}{(K + G_n)^{1/2}} \exp \left\{ y_n^T \alpha^* - 1^T \{b(\alpha^*) - c(y_n)\} - (\alpha^* - \mu)^T G_n (\alpha^* - \mu) / 2 \right\},
\]

where \(\hat{\psi} = (\hat{\gamma}_1, \ldots, \hat{\gamma}_m, \hat{\phi}_1, \ldots, \hat{\phi}_m, \hat{\sigma}_1^2, \ldots, \hat{\sigma}_m^2)\) is the MLE.

Goal: Optimize an objective function over \((m, \tau_1, \ldots, \tau_m)\).

- use minimum description length (MDL) as an objective function
- use genetic algorithm for optimization
Count Data Example

Model: \( Y_t \mid \alpha_t \sim Pois(\exp\{\beta + \alpha_t\}) \), \( \alpha_t = \phi \alpha_{t-1} + \varepsilon_t \), \( \{\varepsilon_t\} \sim \text{IID } N(0, \sigma^2) \)

True model:
- \( Y_t \mid \alpha_t \sim Pois(\exp\{.7 + \alpha_t\}) \), \( \alpha_t = .5 \alpha_{t-1} + \varepsilon_t \), \( \{\varepsilon_t\} \sim \text{IID } N(0, .3) \), \( t < 250 \)
- \( Y_t \mid \alpha_t \sim Pois(\exp\{.7 + \alpha_t\}) \), \( \alpha_t = -.5 \alpha_{t-1} + \varepsilon_t \), \( \{\varepsilon_t\} \sim \text{IID } N(0, .3) \), \( t > 250 \).
- GA estimate 251, time 267secs
SV Process Example

Model: \( Y_t \mid \alpha_t \sim N(0, \exp(\alpha_t)) , \alpha_t = \gamma + \phi \alpha_{t-1} + \varepsilon_t , \{\varepsilon_t\} \sim \text{IID } N(0, \sigma^2) \)

True model:
- \( Y_t \mid \alpha_t \sim N(0, \exp(\alpha_t)) , \alpha_t = -.05 + .975\alpha_{t-1} + \varepsilon_t , \{\varepsilon_t\} \sim \text{IID } N(0, .05) , \ t \leq 750 \)
- \( Y_t \mid \alpha_t \sim N(0, \exp(\alpha_t)) , \alpha_t = -.25 + .900\alpha_{t-1} + \varepsilon_t , \{\varepsilon_t\} \sim \text{IID } N(0, .25) , \ t > 750. \)
- GA estimate 754, time 1053 secs
SV Process Example

Model:  \( Y_t | \alpha_t \sim N(0, \exp(\alpha_t)), \ \alpha_t = \gamma + \phi \alpha_{t-1} + \epsilon_t, \ \{\epsilon_t\} \sim \text{IID } N(0, \sigma^2) \)

True model:
- \( Y_t | \alpha_t \sim N(0, \exp(\alpha_t)), \ \alpha_t = -0.175 + 0.977\alpha_{t-1} + \epsilon_t, \ \{\epsilon_t\} \sim \text{IID } N(0, 0.1810), \ t \leq 250 \)
- \( Y_t | \alpha_t \sim N(0, \exp(\alpha_t)), \ \alpha_t = -0.010 + 0.996\alpha_{t-1} + \epsilon_t, \ \{\epsilon_t\} \sim \text{IID } N(0, 0.0089), \ t > 250. \)
- GA estimate 251, time 269s
True model:

- \( Y_t \mid \alpha_t \sim N(0, \exp\{\alpha_t\}) \), \( \alpha_t = -0.175 + 0.977\alpha_{t-1} + \varepsilon_t \), \( \{\varepsilon_t\} \sim \text{IID } N(0, 0.1810) \), \( t \leq 250 \)
- \( Y_t \mid \alpha_t \sim N(0, \exp\{\alpha_t\}) \), \( \alpha_t = -0.010 + 0.996\alpha_{t-1} + \varepsilon_t \), \( \{\varepsilon_t\} \sim \text{IID } N(0, 0.0089) \), \( t > 250 \).

Fitted model based on no structural break:

- \( Y_t \mid \alpha_t \sim N(0, \exp\{\alpha_t\}) \), \( \alpha_t = -0.0645 + 0.9889\alpha_{t-1} + \varepsilon_t \), \( \{\varepsilon_t\} \sim \text{IID } N(0, 0.0935) \)
SV Process Example-(cont)

Fitted model based on no structural break:

- \( Y_t \mid \alpha_t \sim N(0, \exp\{\alpha_t\}) \), \( \alpha_t = -0.0645 + 0.9889\alpha_{t-1} + \epsilon_t \), \( \{\epsilon_t\}\sim\text{IID } N(0, 0.0935) \)
1. **MDL** appears to be a good criterion for detecting structural breaks.

2. Optimization using a *genetic algorithm* is well suited to find a near optimal value of MDL.

3. This procedure extends easily to *multivariate* problems.

4. While estimating structural breaks for nonlinear time series models is *more challenging*, this paradigm of using **MDL together GA** holds promise for break detection in *parameter-driven* models and other nonlinear models.