Structural Break Detection in Time Series Models

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.
How many segments do you see?

$\tau_1 = 51 \quad \tau_2 = 151 \quad \text{time} \quad \tau_3 = 251$
Illustrative Example

Auto-PARM=Auto-Piecewise AutoRegressive Modeling

4 pieces, 2.58 seconds.

\[ \tau_1 = 51 \quad \tau_2 = 157 \quad \text{time } \tau_3 = 259 \]
Introduction
  - Examples
    - AR
    - GARCH
    - Stochastic volatility
    - State space models

Model selection using Minimum Description Length (MDL)
  - General principles
  - Application to AR models with breaks

Optimization using a Genetic Algorithm
  - Basics
  - Implementation for structural break estimation

Simulation results

Applications

Simulation results for GARCH and SSM
Examples

1. Piecewise AR model:

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

where \( \tau_0 = 1 < \tau_1 < \ldots < \tau_{m-1} < \tau_m = n + 1, \) 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
Piecewise AR models (cont)

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.
Motivation for using piecewise AR models:

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

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

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 \( \{\tilde{Y}_{t,n}\} \) with

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

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 \( u \).
Examples (cont)

2. Segmented GARCH model:

\[ Y_t = \sigma_t \varepsilon_t, \]
\[ \sigma_t^2 = \omega_j + \alpha_{j_1} Y_{t-1}^2 + \cdots + \alpha_{j_p} Y_{t-p_j}^2 + \beta_{j_1} \sigma_{t-1}^2 + \cdots + \beta_{j_q} \sigma_{t-q_j}^2, \quad \text{if } \tau_{j-1} \leq t < \tau_j, \]

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

3. Segmented stochastic volatility model:

\[ Y_t = \sigma_t \varepsilon_t, \]
\[ \log \sigma_t^2 = \gamma_j + \phi_{j_1} \log \sigma_{t-1}^2 + \cdots + \phi_{j_p} \log \sigma_{t-p_j}^2 + \nu_j \eta_t, \quad \text{if } \tau_{j-1} \leq t < \tau_j. \]

4. Segmented state-space model (SVM a special case):

\[ p(y_t \mid \alpha_t, \ldots, \alpha_1, y_{t-1}, \ldots, y_1) = p(y_t \mid \alpha_t) \text{ is specified} \]
\[ \alpha_t = \gamma_j + \phi_{j_1} \alpha_{t-1} + \cdots + \phi_{j_p} \alpha_{t-p_j} + \sigma_j \eta_t, \quad \text{if } \tau_{j-1} \leq t < \tau_j. \]
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

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

\[ L(\hat{e}|\hat{F}) = \text{code length of the residuals based on the fitted model} \]
Model Selection Using Minimum Description Length (cont)

Applied to the segmented AR model:

\[ Y_t = \gamma_j + \phi_{j_1} Y_{t-1} + \cdots + \phi_{j_p} Y_{t-p} + \sigma_j \epsilon_t, \text{ if } \tau_{j-1} \leq t < \tau_j, \]

First term \( L(\hat{F}_y) \): 

\[
L(\hat{F}_y) = L(m) + L(\tau_1, \ldots, \tau_m) + L(p_1, \ldots, p_m) + L(\hat{\psi}_1 | y) + \cdots + L(\hat{\psi}_m | y) \\
= \log_2 m + m \log_2 n + \sum_{j=1}^{m} \log_2 p_j + \sum_{j=1}^{m} \frac{p_j + 2}{2} \log_2 n_j
\]

Encoding:

- integer \( I \) : \( \log_2 I \) bits (if \( I \) unbounded)  
- \( \log_2 I_U \) bits (if \( I \) bounded by \( I_U \))

MLE \( \hat{\theta} \) : \( \frac{1}{2} \log_2 N \) bits (where \( N \) = number of observations used to compute \( \hat{\theta} \); Rissanen (1989))
Second term $L(\hat{e} | \hat{F})$ : Using Shannon’s classical results on information theory, Rissanen demonstrates that the code length of can be approximated by the negative of the log-likelihood of the fitted model, i.e., by

$$L(\hat{e} | \hat{F}) \approx -\sum_{j=1}^{m} \log_2 L(\hat{\psi}_j | y)$$

For fixed values of $m$, $(\tau_1, p_1), \ldots, (\tau_m, p_m)$, we define the MDL as

$$MDL(m, (\tau_1, p_1), \ldots, (\tau_m, p_m))$$

$$= \log_2 m + m \log_2 n + \sum_{j=1}^{m} \log_2 p_j + \sum_{j=1}^{m} \frac{p_j + 2}{2} \log_2 n_j - \sum_{j=1}^{m} \log_2 L(\hat{\psi}_j | y)$$

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 for AR processes, we use Y-W estimates of AR parameters and we replace

$$-\log_2 L(\hat{\psi}_j | y) \quad \text{with} \quad \log_2 (2\pi \hat{\sigma}_j^2) + n_j$$
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

$$(m,(\tau_1, p_1),\ldots,(\tau_m, p_m)) \rightarrow c = (\delta_1,\ldots,\delta_n),$$

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,

$$c = (2, -1, -1, -1, -1, 0, -1, -1, -1, 0, -1, -1, 3, -1, -1, -1, -1, -1, -1, -1, -1)$$

$t$: 1 6 11 15

would correspond to a process as follows:

AR(2), $t=1:5$; AR(0), $t=6:10$; AR(0), $t=11:14$; 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 $p$ w.p. $1 - \pi_1 - \pi_2$. 
Implementation of Genetic Algorithm—(cont)

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.
Simulation Examples-based on Ombao et al. (2001) test cases

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)

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>0.9945</td>
<td></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>
2. Slowly varying AR(2) model:

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

where \( a_t = .8[1 - 0.5\cos(\pi t / 1024)] \), and \( \{\varepsilon_t\} \sim \text{IID } N(0,1) \).
2. 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>
2. 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.
3. Piecewise ARMA:

\[
Y_t = \begin{cases} 
-0.9Y_{t-1} + \varepsilon_t + 0.7\varepsilon_{t-1}, & \text{if } 1 \leq t < 513, \\
0.9Y_{t-1} + \varepsilon_t, & \text{if } 513 \leq t < 769, \\
\varepsilon_t - 0.7\varepsilon_{t-1}, & \text{if } 769 \leq t \leq 1024.
\end{cases}
\]

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

GA results: 3 pieces, breaks at $\tau_1 = 513$, $\tau_2 = 769$. Total run time 1.53 secs

Fitted model: AR orders 4, 1, 2
Consistency.

Suppose the number of change points $m$ is known and let

$$\lambda_1 = \frac{\tau_1}{n}, \ldots, \lambda_m = \frac{\tau_m}{n}$$

be the relative (true) changepoints. Then

$$\hat{\lambda}_j \to \lambda_j \quad \text{a.s.}$$

where $\hat{\lambda}_j = \frac{\hat{\tau}_j}{n}$ and $\hat{\tau}_j = \text{Auto-PARM estimate of } \tau_j$. 
**Application to GARCH (cont)**

**Garch(1,1) model:** \( Y_t = \sigma_t \varepsilon_t, \quad \{\varepsilon_t\} \sim \text{IID}(0,1) \)

\[ \sigma_t^2 = \omega_j + \alpha_j Y_{t-1}^2 + \beta_j \sigma_{t-1}^2, \quad \text{if } \tau_{j-1} \leq t < \tau_j. \]

\[ \sigma_t^2 = \begin{cases} .4 + .1 Y_{t-1}^2 + .5 \sigma_{t-1}^2, & \text{if } 1 \leq t < 501, \\ .4 + .1 Y_{t-1}^2 + .6 \sigma_{t-1}^2, & \text{if } 501 \leq t < 1000 \end{cases} \]

**CP estimate = 506**

**AG = Andreou and Ghysels (2002)**

<table>
<thead>
<tr>
<th># of CPs</th>
<th>GA %</th>
<th>AG %</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>80.4</td>
<td>72.0</td>
</tr>
<tr>
<td>1</td>
<td>19.2</td>
<td>24.0</td>
</tr>
<tr>
<td>≥ 2</td>
<td>0.4</td>
<td>0.4</td>
</tr>
</tbody>
</table>
Application to GARCH (cont)

More simulation results for Garch(1,1): \( Y_t = \sigma_t \varepsilon_t, \quad \{\varepsilon_t\} \sim \text{IID}(0,1) \)

\[
\sigma_t^2 = \begin{cases} 
0.05 + 0.4Y_{t-1}^2 + 0.3\sigma_{t-1}^2, & \text{if } 1 \leq t < \tau_1, \\
1.00 + 0.3Y_{t-1}^2 + 0.2\sigma_{t-1}^2, & \text{if } \tau_1 \leq t < 1000
\end{cases}
\]

<table>
<thead>
<tr>
<th>( \tau_1 )</th>
<th>Mean</th>
<th>SE</th>
<th>Med</th>
<th>Freq</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>GA</td>
<td>52.62</td>
<td>11.70</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>Berkes</td>
<td>71.40</td>
<td>12.40</td>
<td>71</td>
</tr>
<tr>
<td>250</td>
<td>GA</td>
<td>251.18</td>
<td>4.50</td>
<td>250</td>
</tr>
<tr>
<td></td>
<td>Berkes</td>
<td>272.30</td>
<td>18.10</td>
<td>271</td>
</tr>
<tr>
<td>500</td>
<td>GA</td>
<td>501.22</td>
<td>4.76</td>
<td>502</td>
</tr>
<tr>
<td></td>
<td>Berkes</td>
<td>516.40</td>
<td>54.70</td>
<td>538</td>
</tr>
</tbody>
</table>

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:

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

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

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

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

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

True model:
- \[ Y_t \mid \alpha_t \sim N(0, \exp\{\alpha_t\}), \ \alpha_t = -0.05 + 0.975\alpha_{t-1} + \epsilon_t, \ \{\epsilon_t\} \sim \text{IID } N(0, 0.05), \ t \leq 750 \]
- \[ Y_t \mid \alpha_t \sim N(0, \exp\{\alpha_t\}), \ \alpha_t = -0.25 + 0.900\alpha_{t-1} + \epsilon_t, \ \{\epsilon_t\} \sim \text{IID } N(0, 0.25), \ t > 750. \]
- GA estimate 754, time 1053 secs
SV Process Example

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

True model:
- \( Y_t | \alpha_t \sim N(0, \exp\{\alpha_t\}), \alpha_t = -.175 + .977\alpha_{t-1} + \varepsilon_t, \{\varepsilon_t\} \sim \text{IID } N(0, .1810), \ t \leq 250 \)
- \( Y_t | \alpha_t \sim N(0, \exp\{\alpha_t\}), \alpha_t = -.010 + .996\alpha_{t-1} + \varepsilon_t, \{\varepsilon_t\} \sim \text{IID } N(0, .0089), \ t > 250. \)
- GA estimate 251, time 269s
SV Process Example-(cont)

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

Fitted model based on no structural break:
- \( Y_t \mid \alpha_t \sim N(0, \exp\{\alpha_t\}) \), \( \alpha_t = -0.0645 + 0.9889\alpha_{t-1} + \epsilon_t , \{\epsilon_t\} \sim \text{IID } N(0, 0.0935) \)
Fitted model based on no structural break:

- \( Y_t \mid \alpha_t \sim N(0, \exp\{\alpha_t\}) \), \( \alpha_t = -0.0645 + 0.9889\alpha_{t-1} + \epsilon_t \), \( \{\epsilon_t\} \sim \text{IID } N(0, 0.0935) \)
Summary Remarks

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.