Detection of Structural Breaks and Outliers in Time Series

Richard A. Davis
Columbia University

Collaborators:

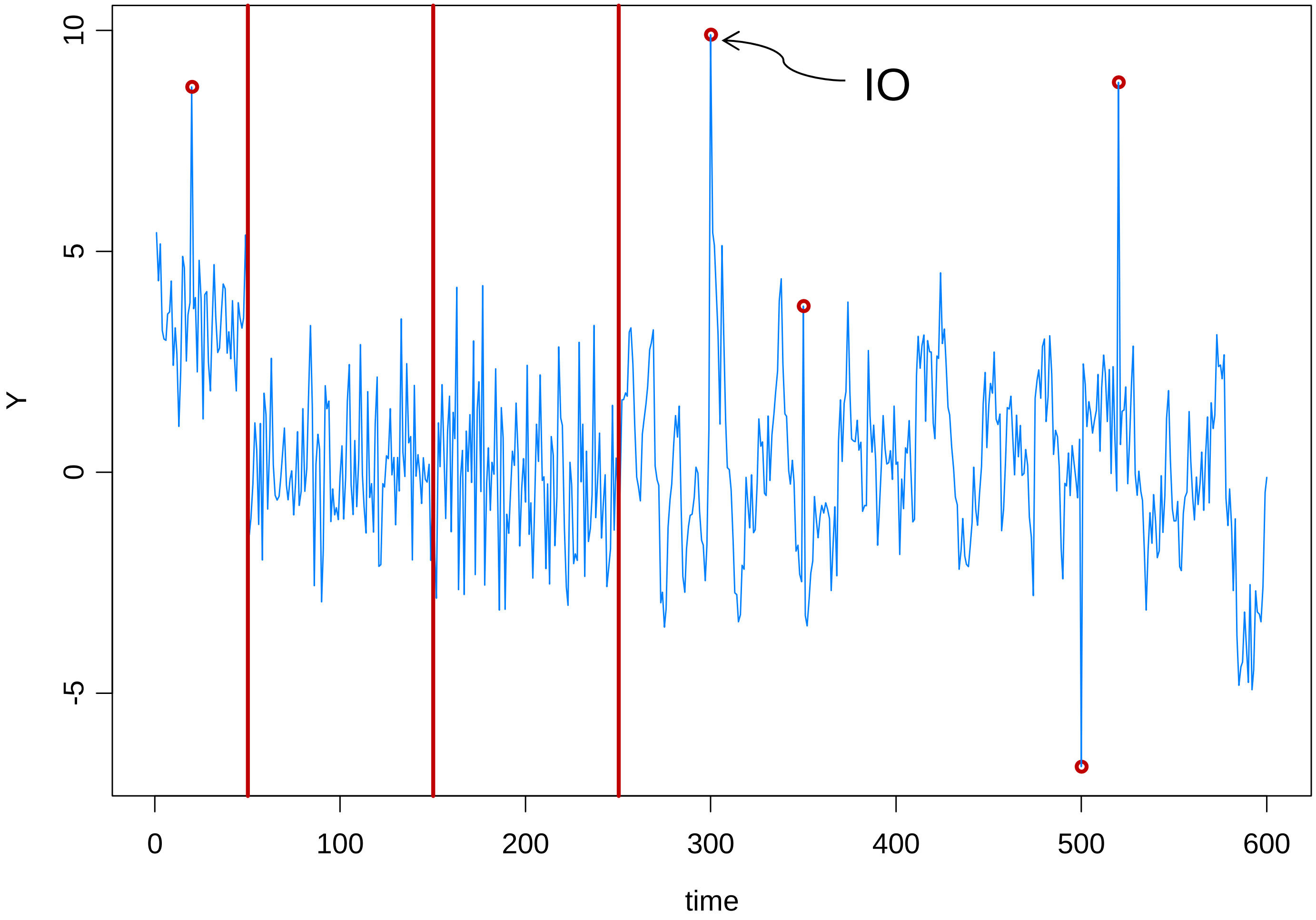
Implementation of AutoPARM/AutoPARMO:
Thomas Lee, UC Davis
Gabriel Rodriguez-Yam, Universida Autónoma Chapingo

Theory behind AutoPARM:
Stacey Hancock, UC Irvine (to be).
Yi-Ching Yao, Academia Sinica

General theory:
Chun Yip Yau, Chinese University of Hong Kong

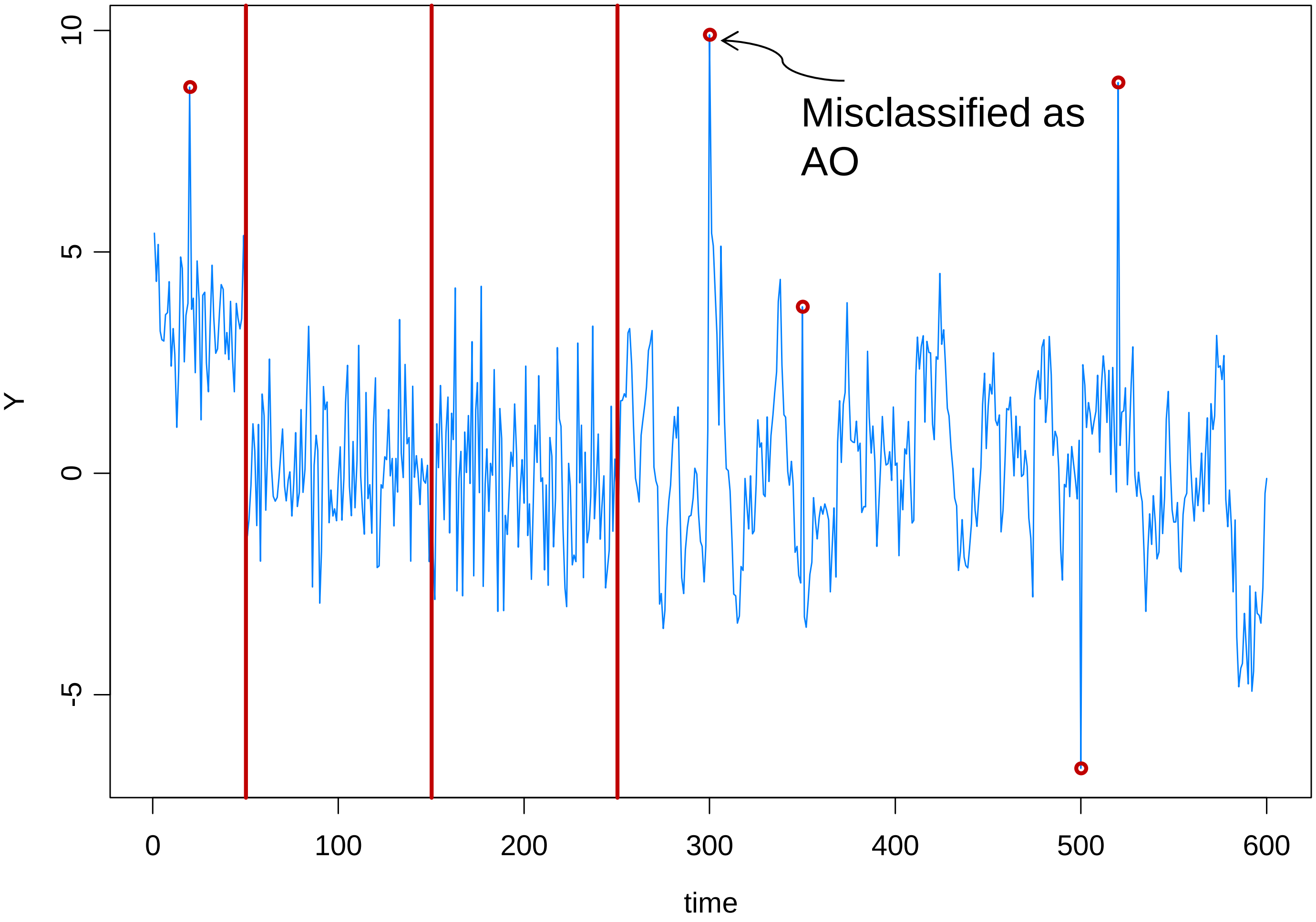
Illustrative Example

Can you spot any unusual observations?



Illustrative Example

Auto-PARMO=Auto-Piecewise AutoRegressive Modeling Outlier

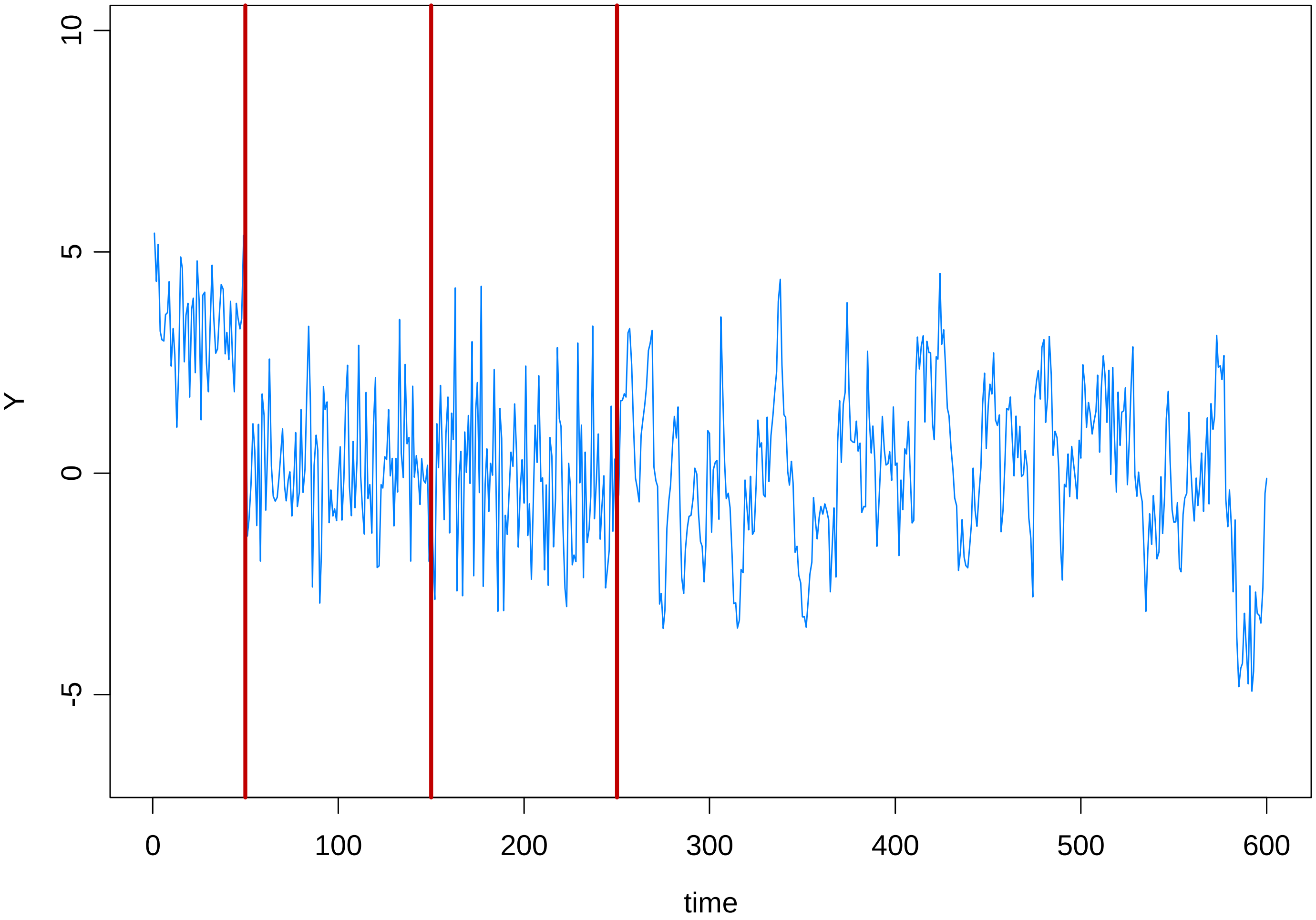


5 AOs
0 IOs

$\tau_1 = 51$ $\tau_2 = 151$ $\tau_3 = 252$

Illustrative Example

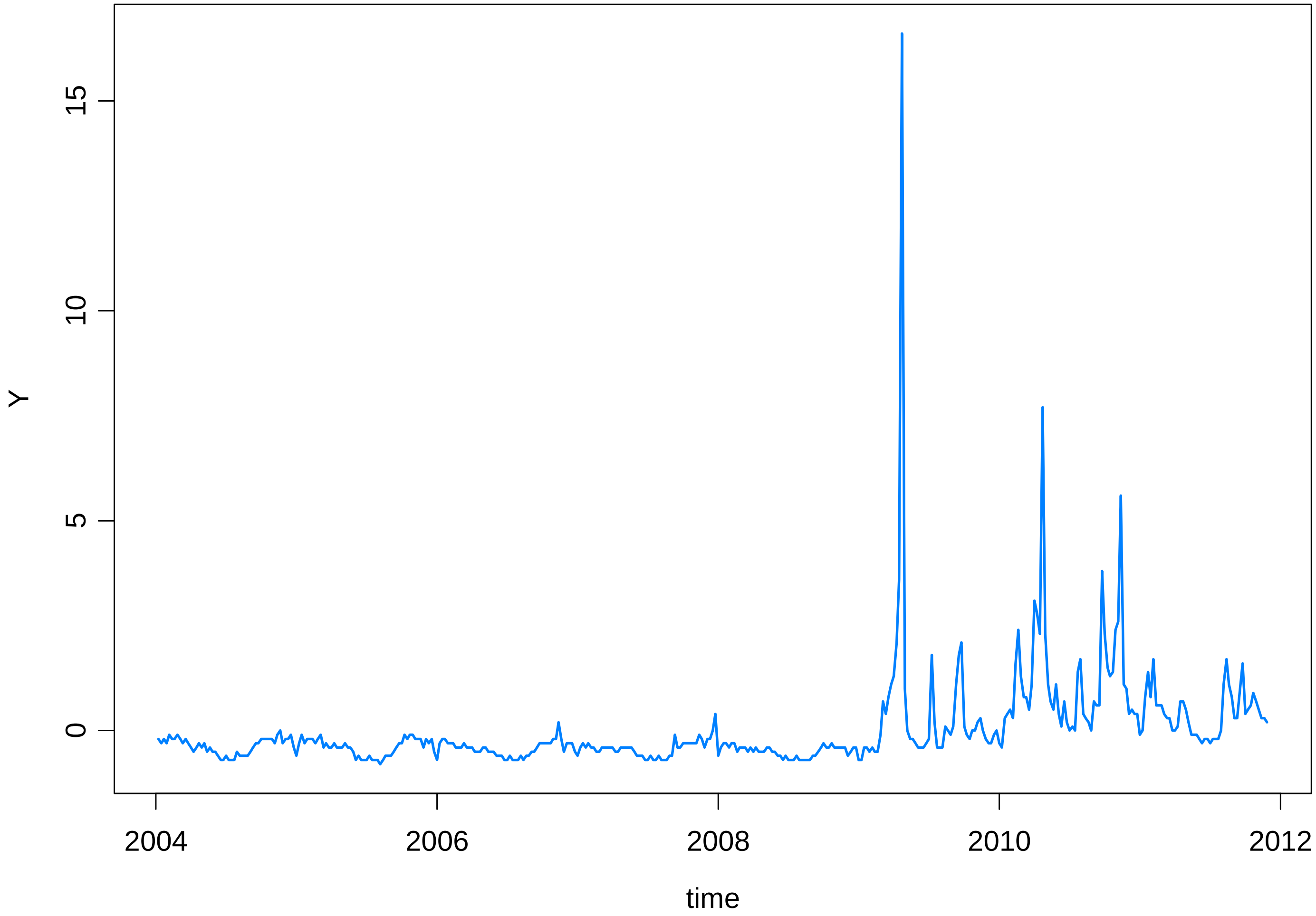
The series w/o any outlier effects—structural breaks easier to identify



A Second Example

Weekly data: January 2004--December 2011.

Any breaks? Outliers?



Game Plan

- Introduction
- Piecewise Models (AR, GARCH, SV, State-space)
- Model selection using Minimum Description Length (MDL)
  - General principles
  - Application to AR models with breaks
- Optimization using a Genetic Algorithm
  - Basics
- Simulation Examples
- Applications
- Comments about theory for AutoPARM
  - Consistency (FLIL)
- AutoPARM with Outliers

Piecewise Models—examples

1. Piecewise AR model:

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

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

Goal: Estimate

m = number of change-points

τ_j = location of jth break point

γ_j = level in jth segment

p_j = order of AR process in jth segment

$(\phi_{j1}, \ldots, \phi_{jp_j})$ = AR coefficients in jth segment

σ_j = scale in jth segment

Examples (cont)

2. Segmented GARCH model:

$$Y_t = \sigma_t \varepsilon_t,$$
$$\sigma_t^2 = \omega_j + \alpha_{j1} Y_{t-1}^2 + \cdots + \alpha_{jp_j} Y_{t-p_j}^2 + \beta_{j1}\sigma_{t-1}^2 + \cdots + \beta_{jq_j}\sigma_{t-q_j}^2, \quad \text{if } \tau_{j\text{-}1} \le t < \tau_j,$$

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

3. Segmented stochastic volatility model:

$$Y_t = \sigma_t \varepsilon_t,$$
$$\log\sigma_t^2 = \gamma_j + \phi_{j1}\log\sigma_{t-1}^2 + \cdots + \phi_{jp_j}\log\sigma_{t-p_j}^2 + \nu_j \eta_t, \quad \text{if } \tau_{j\text{-}1} \le t < \tau_j.$$

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

$$p(y_t \mid \alpha_t, \ldots, \alpha_1, y_{t-1}, \ldots, y_1) = p(y_t \mid \alpha_t) \text{ is specified}$$
$$\alpha_t = \gamma_j + \phi_{j1}\alpha_{t-1} + \cdots + \phi_{jp_j}\alpha_{t-p_j} + \sigma_j \eta_t, \quad \text{if } \tau_{j\text{-}1} \le t < \tau_j.$$

Model Selection Using Minimum Description Length

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).

$\boldsymbol{M}$ = class of operating models for $y = (y_1, \ldots, y_n)$

$L_{\boldsymbol{F}}(y)$ = code length of y relative to $\boldsymbol{F} \in \boldsymbol{M}$

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

$$L_{\boldsymbol{F}}(y) = L(\hat{\boldsymbol{F}}|y) + L(\hat{e}\,|\,\hat{\boldsymbol{F}}),$$

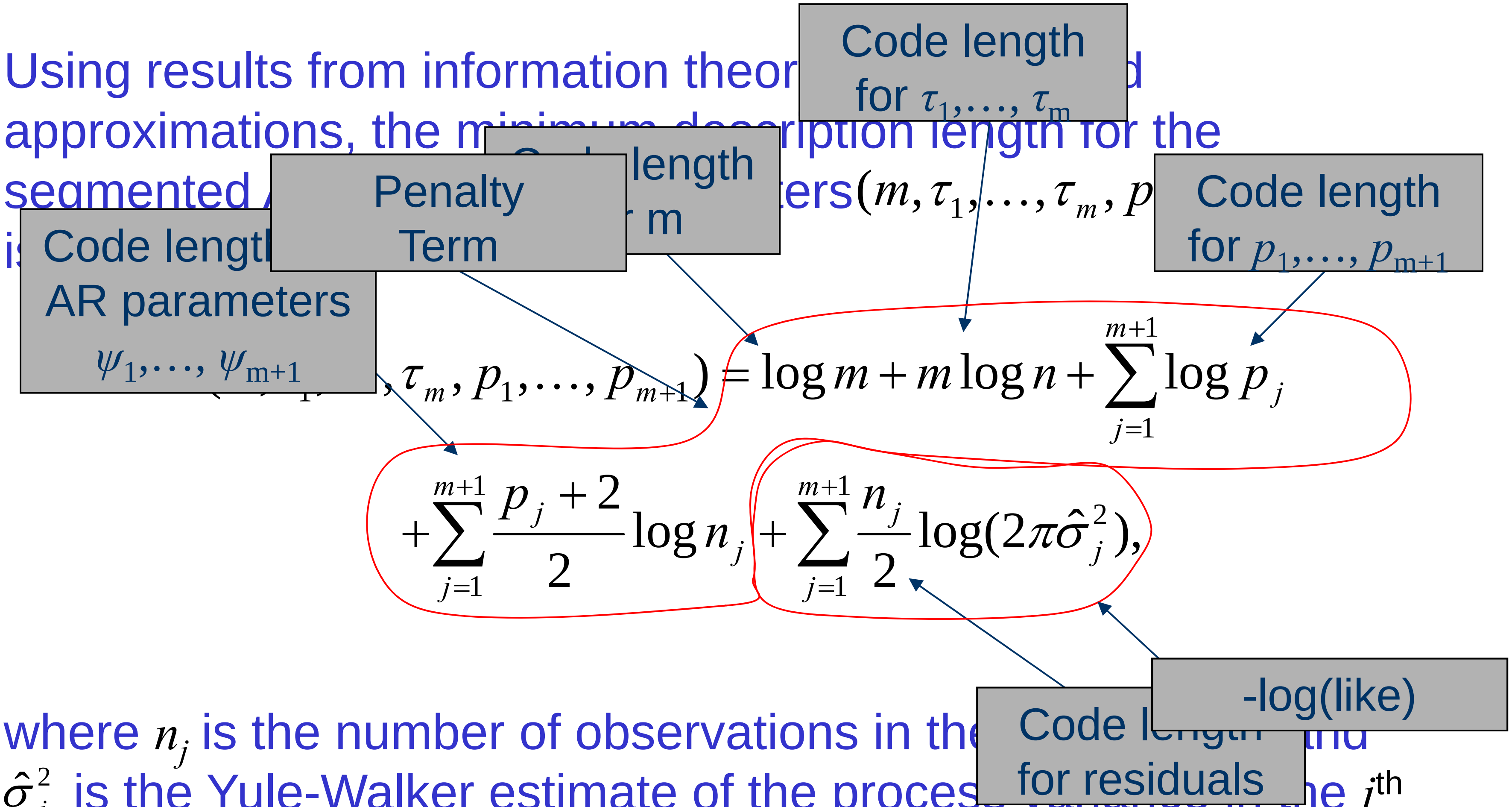
where

$L(\hat{\boldsymbol{F}}|y)$ = code length of the fitted model for $\boldsymbol{F}$

$L(\hat{e}|\hat{\boldsymbol{F}})$ = code length of the residuals based on the fitted model

Model Selection Using Minimum Description Length (cont)

Using results from information theor d approximations, the m scription length for the segmented A ters$(m,\tau_1,\ldots,\tau_m,p$ i



$$\ldots,\tau_m,p_1,\ldots,p_{m+1}) = \log m + m\log n + \sum_{j=1}^{m+1}\log p_j + \sum_{j=1}^{m+1}\frac{p_j+2}{2}\log n_j + \sum_{j=1}^{m+1}\frac{n_j}{2}\log(2\pi\hat{\sigma}_j^2),$$

Code length for $\tau_1,\ldots,\tau_m$

Code length for m

Penalty Term

Code length for $p_1,\ldots,p_{m+1}$

Code length AR parameters $\psi_1,\ldots,\psi_{m+1}$

-log(like)

Code length for residuals

where n_j is the number of observations in the and $\hat{\sigma}_j^2$ is the Yule-Walker estimate of the proces variance in the j^{th} segment.

Minimizing the MDL

- Select the best-fitting model for the data as the model that minimizes the minimum description length with respect to the number of change-points m, the change-point locations, $\tau_1, \ldots, \tau_m$, and the AR orders $p_1, \ldots, p_{m+1}$.
- The dependence of the minimum description length on the autoregressive coefficient parameter estimates is only through the white noise estimates $\hat{\sigma}_j^2$, $j = 1, \ldots, m + 1$.
- Numerical minimization carried out using a genetic algorithm, which mimics natural evolution (see Davis et al., 2006).

Optimization Using Genetic Algorithm

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

$$(m,(\tau_1,p_1)\ldots,(\tau_m,p_m)) \longleftrightarrow c=(\delta_1,\ldots,\delta_{\mathrm{n}}),$$

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, -1, 0, -1, -1, -1, 3, -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 π_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. ½ and $\delta_{j,k}$ w.p. ½
- 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. π_1 ; -1 w.p. π_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.

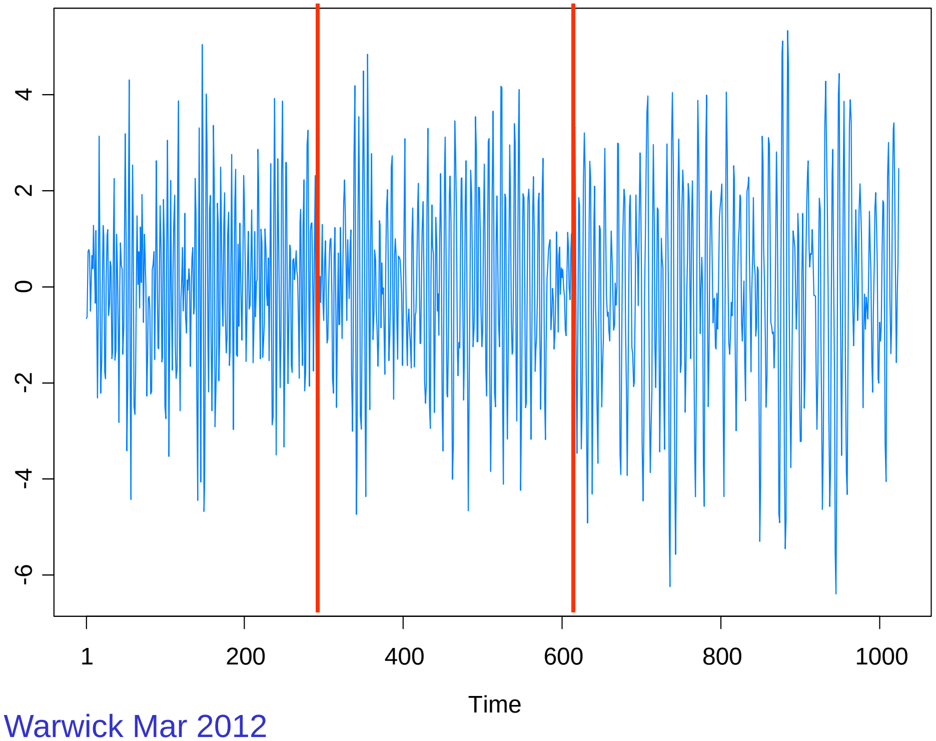
- ➢ use multiple *islands*, in which populations run independently, and then allow *migration* after a fixed number of generations. This implementation is amenable to *parallel computing*.

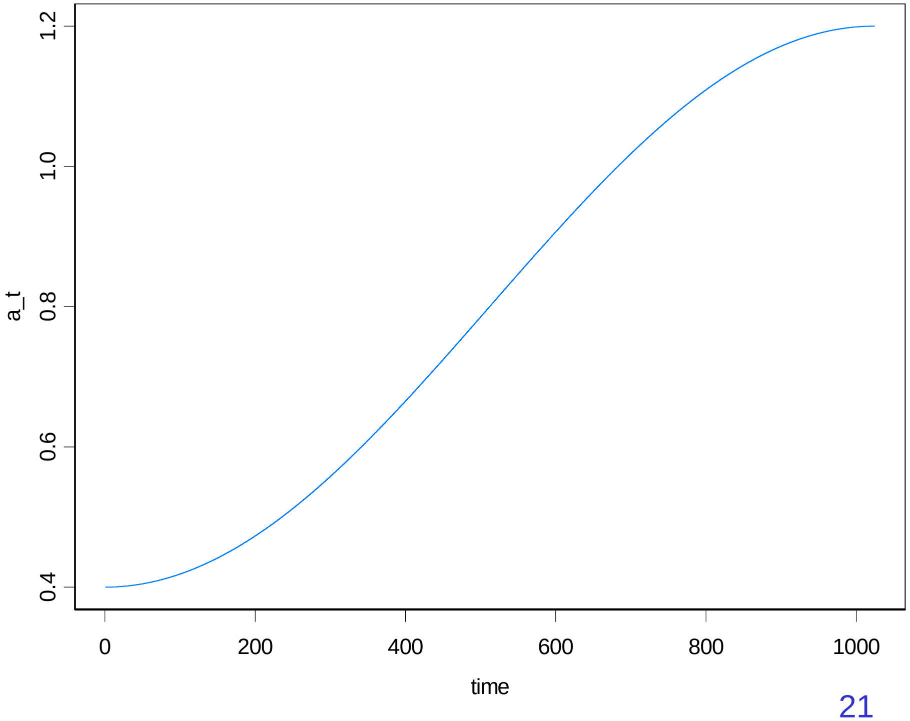
Simulation Examples (cont)

2. Slowly varying AR(2) model:

$$Y_t = a_t Y_{t-1} - .81 Y_{t-2} + \varepsilon_t \quad \text{if } 1 \le t \le 1024$$

where $a_t = .8[1 - 0.5\cos(\pi t / 1024)]$, and $\{\varepsilon_t\}$ ~ IID N(0,1).





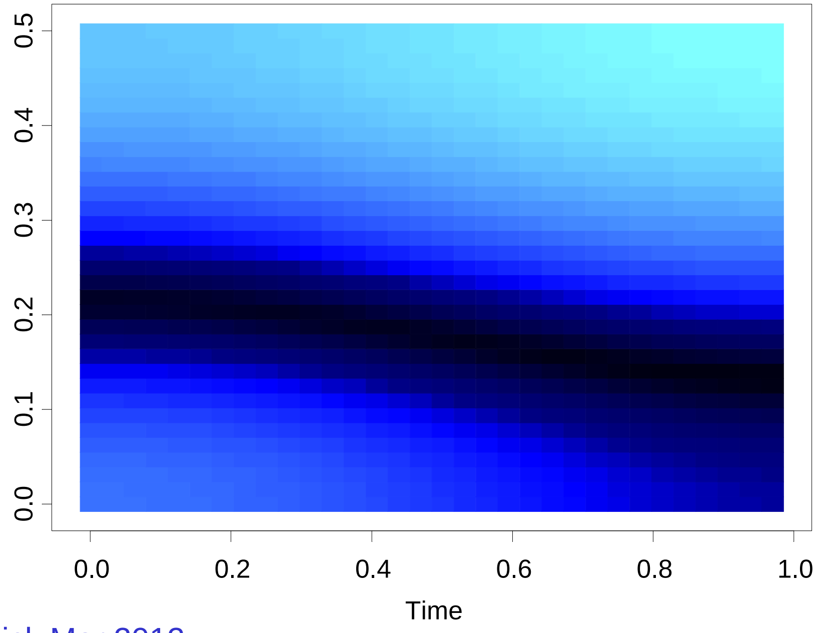
2. Slowly varying AR(2) (cont)

GA results: 3 pieces, breaks at τ_1=293, τ_2=615. Total run time 27.45 secs

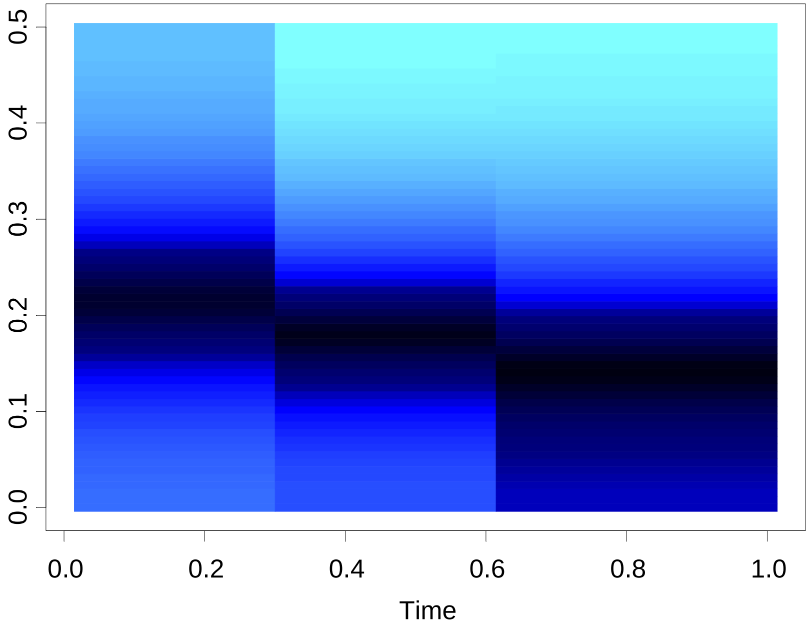
Fitted model:

	ϕ_1	ϕ_2	σ^2
1- 292:	.365	-0.753	1.149
293- 614:	.821	-0.790	1.176
615-1024:	1.084	-0.760	0.960

True Model

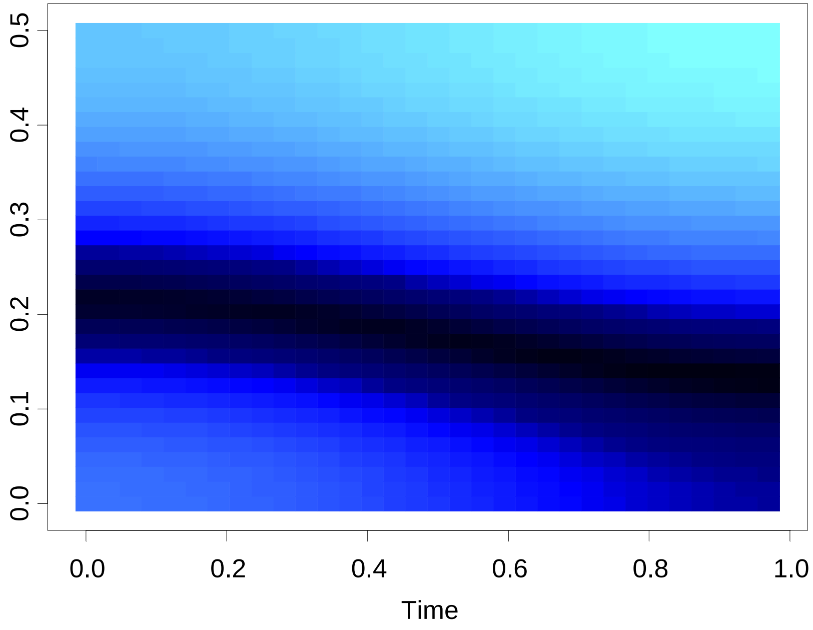


Fitted Model

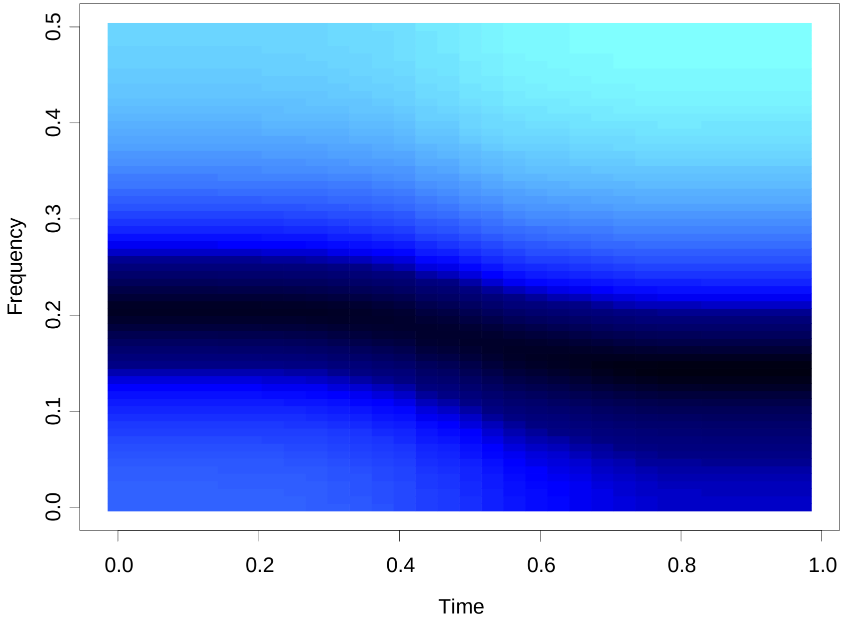


2. Slowly varying AR(2) (cont)

In the graph below right, we average the spectogram over the *GA fitted models* generated from each of the 200 simulated realizations.



True Model



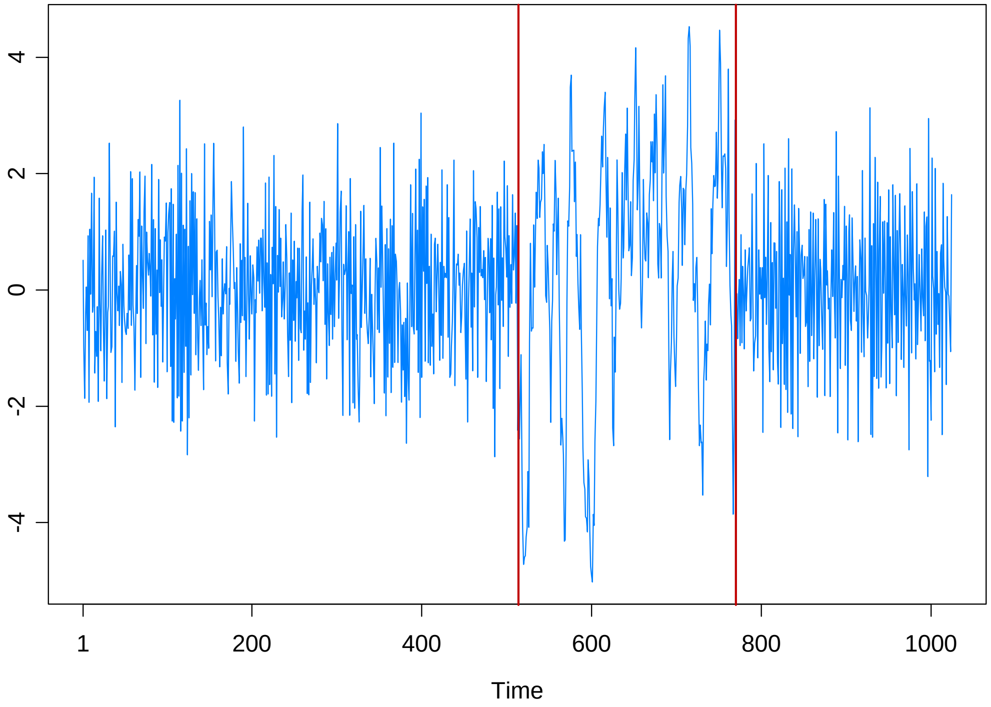
Average Model

Simulation Examples (cont)

3. Piecewise ARMA:

$$Y_t = \begin{cases} -.9Y_{t-1} + \varepsilon_t + .7\varepsilon_{t-1}, & \text{if } 1 \le t < 513, \\ .9Y_{t-1} + \varepsilon_t, & \text{if } 513 \le t < 769, \\ \varepsilon_t - .7\varepsilon_{t-1}, & \text{if } 769 \le t \le 1024. \end{cases}$$

where $\{\varepsilon_t\}$ ~ IID N(0,1).

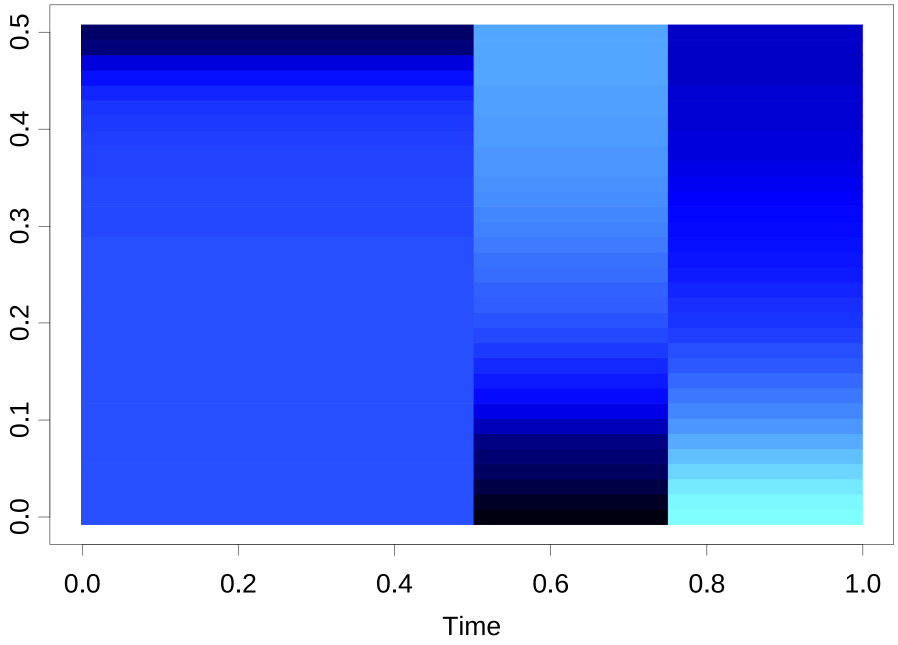


3. Piecewise ARMA (cont)

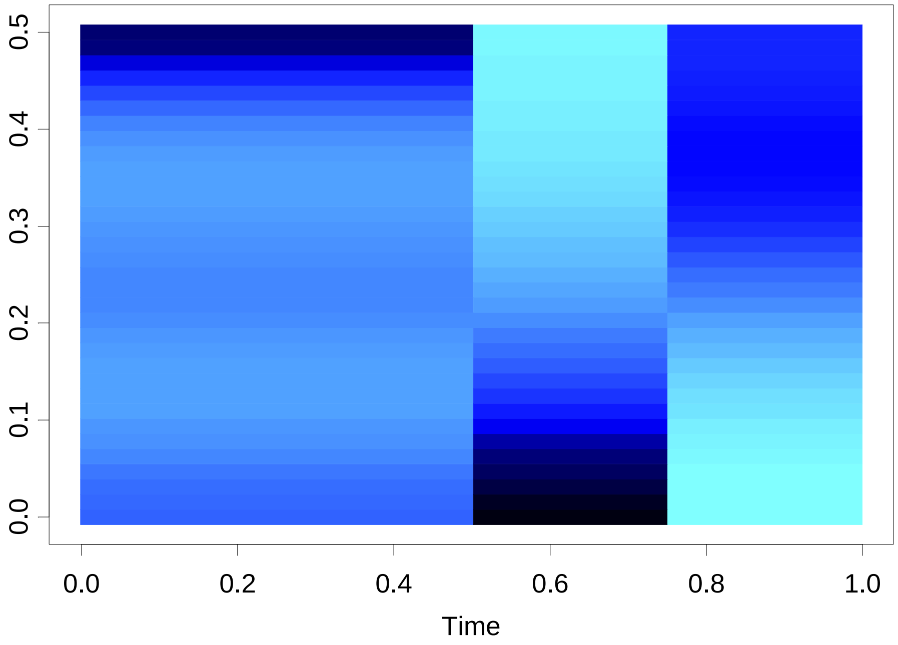
GA results: 3 pieces, breaks at τ_1=513, τ_2=769. Total run time 1.53 secs

Fitted model: AR orders 4, 1, 2

True Model



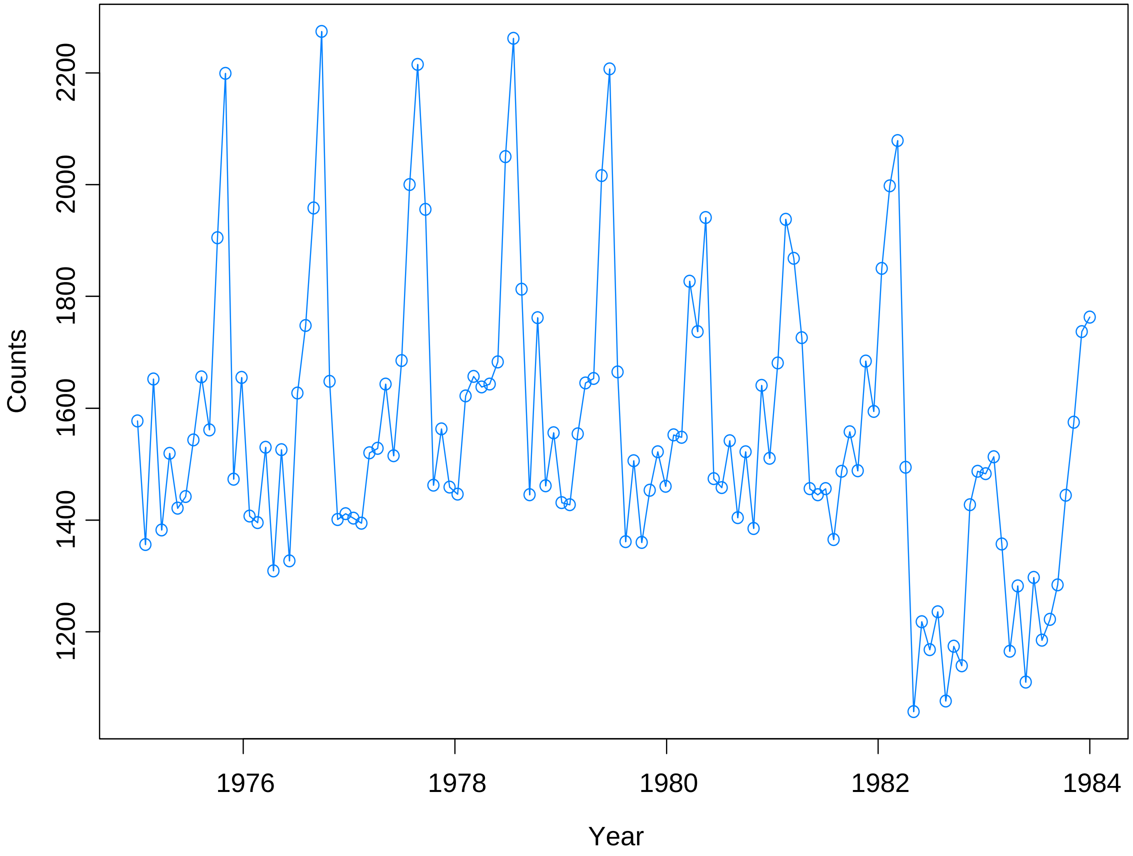
Fitted Model



Example--Monthly Deaths & Serious Injuries, UK

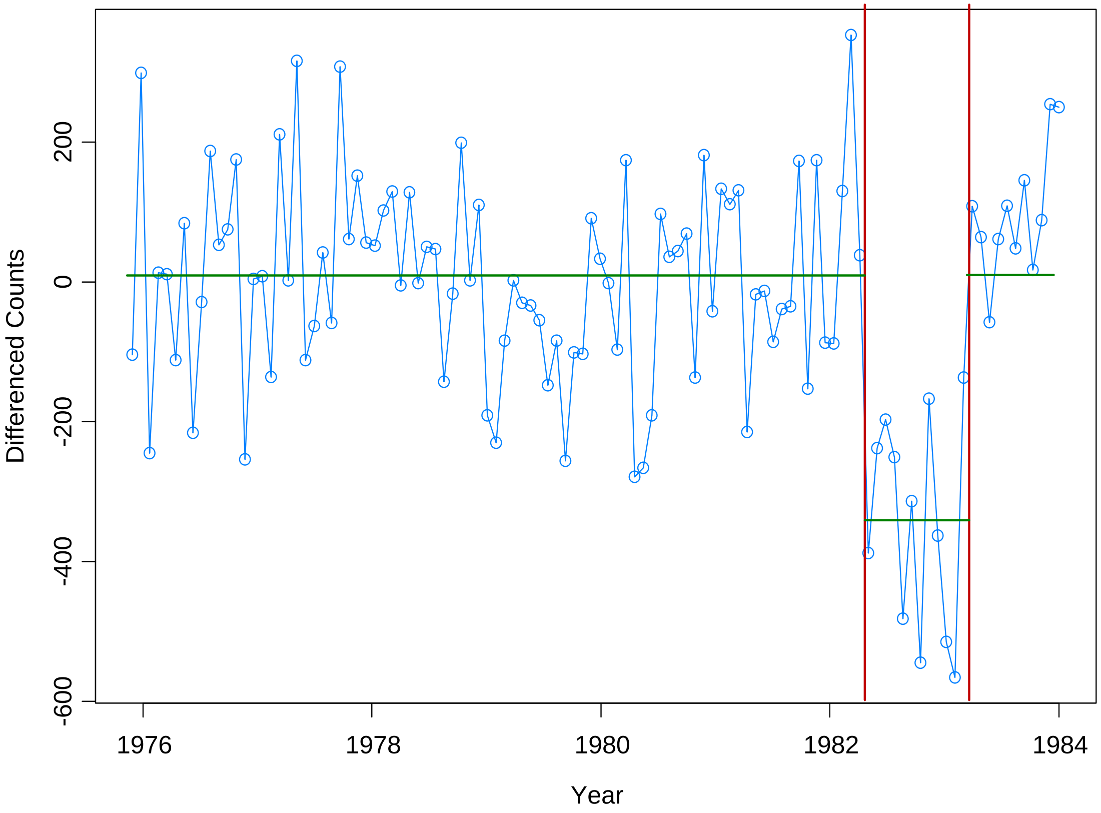
Data: y_t = number of monthly deaths and serious injuries in UK, Jan \`75 – Dec \`84, (t = 1,…, 120)

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,…, 120)
Remark: Seat belt legislation introduced in Feb `83 (t = 99).



Traditional regression analysis:

$$Y_t = a + bf(t) + W_t,$$

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

$$X_t = Y_t - Y_{t-12}$$
$$= bg(t) + N_t$$

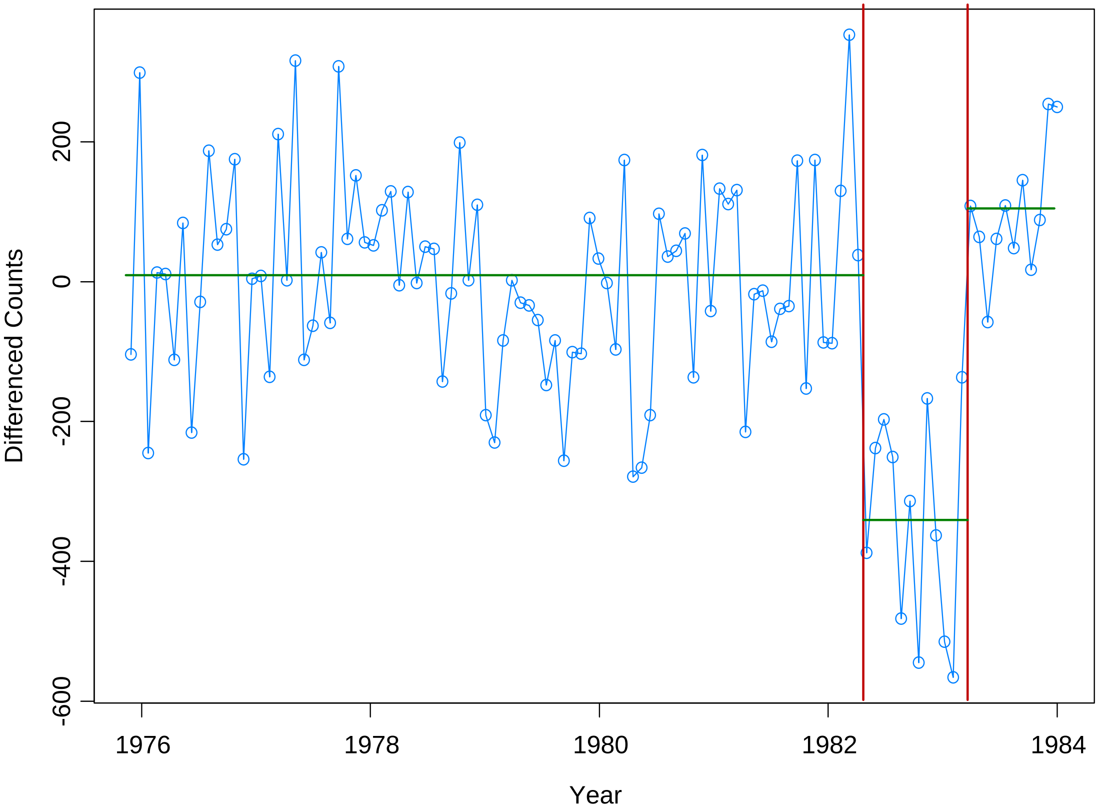
$$g(t) = \begin{cases} 1, & \text{if } 99 \le t \le 110, \\ 0, & \text{otherwise.} \end{cases}$$

Model: b=–373.4, $\{N_t\}$~AR(13).

Example: Monthly Deaths & Serious Injuries, UK

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

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



Results from GA: 3 pieces; time = 4.4secs

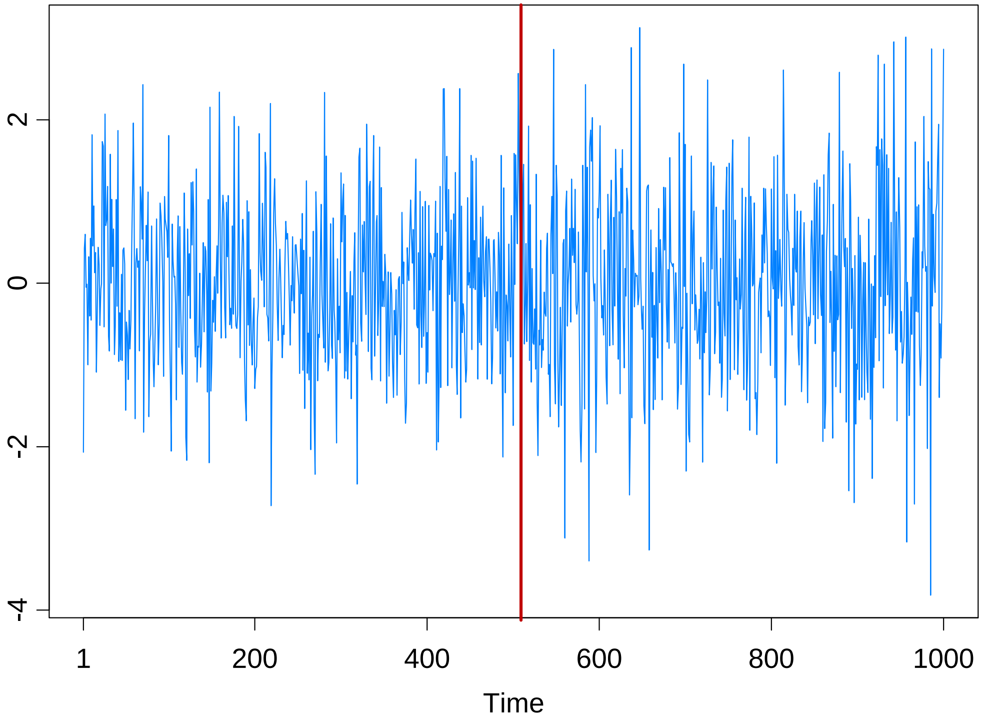
Piece 1: (t=1,…, 98) IID; Piece 2: (t=99,…108) IID; Piece 3: t=109,…,120 AR(1)

Application to GARCH

Garch(1,1) model:

$$Y_t = \sigma_t \varepsilon_t, \qquad \{\varepsilon_t\} \sim \text{IID}(0,1)$$

$$\sigma_t^2 = \omega_j \; + \alpha_j Y_{t-1}^2 + \beta_j \sigma_{t-1}^2, \qquad \text{if } \tau_{j\text{-}1} \le t < \tau_j.$$



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

# of CPs	GA %	AG %
0	80.4	72.0
1	19.2	24.0
≥ 2	0.4	0.4

CP estimate = 506

AG = Andreou and Ghysels (2002)

Application to GARCH (cont)

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

$$\sigma_t^2 = \begin{cases} .05 + .4Y_{t-1}^2 + .3\sigma_{t-1}^2, & \text{if } 1 \le t < \tau_1, \\ 1.00 + .3Y_{t-1}^2 + .2\sigma_{t-1}^2, & \text{if } \tau_1 \le t < 1000. \end{cases}$$

τ_1		Mean	SE	Med	Freq
50	GA Berkes	**52.62** **71.40**	11.70 12.40	**50** **71**	.98
250	GA Berkes	**251.18** **272.30**	4.50 18.10	**250** **271**	.99
500	GA Berkes	**501.22** **516.40**	4.76 54.70	**502** **538**	.98

Berkes = Berkes, Gombay, Horvath, and Kokoszka (2004).

Application to Parameter-Driven SS Models

State Space Model Setup:

Observation equation:

$$p(y_t \mid \alpha_t) = \exp\{\alpha_t y_t - b(\alpha_t) + c(y_t)\}.$$

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

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

where $1 = \tau_0 < \tau_1 < \dots < \tau_m < n$, and $\{\varepsilon_t\}$ ~ IID N(0,1).

Parameters:

m = number of break points

τ_k = location of break points

γ_k = level in kth epoch

ϕ_k = AR coefficients kth epoch

σ_k = scale in kth epoch

Application to Structural Breaks—(cont)

Estimation: For (m, τ_1, . . . , τ_m) fixed, calculate the approximate likelihood evaluated at the "MLE", i.e.,

$$L_a(\hat{\psi};\mathrm{y}_\mathrm{n}) = \frac{|G_n|^{1/2}}{(K+G_n)^{1/2}}\exp\{\mathrm{y}_n^T\alpha^* - 1^T\{b(\alpha^*) - c(\mathrm{y}_n)\} - (\alpha^* - \mu)^T G_n(\alpha^* - \mu)/2\},$$

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.

Remark: The exact likelihood is given by the following formula

$$L(\psi;\mathrm{y}_n) = L_a(\psi;\mathrm{y}_n)Er_a(\psi),$$

where

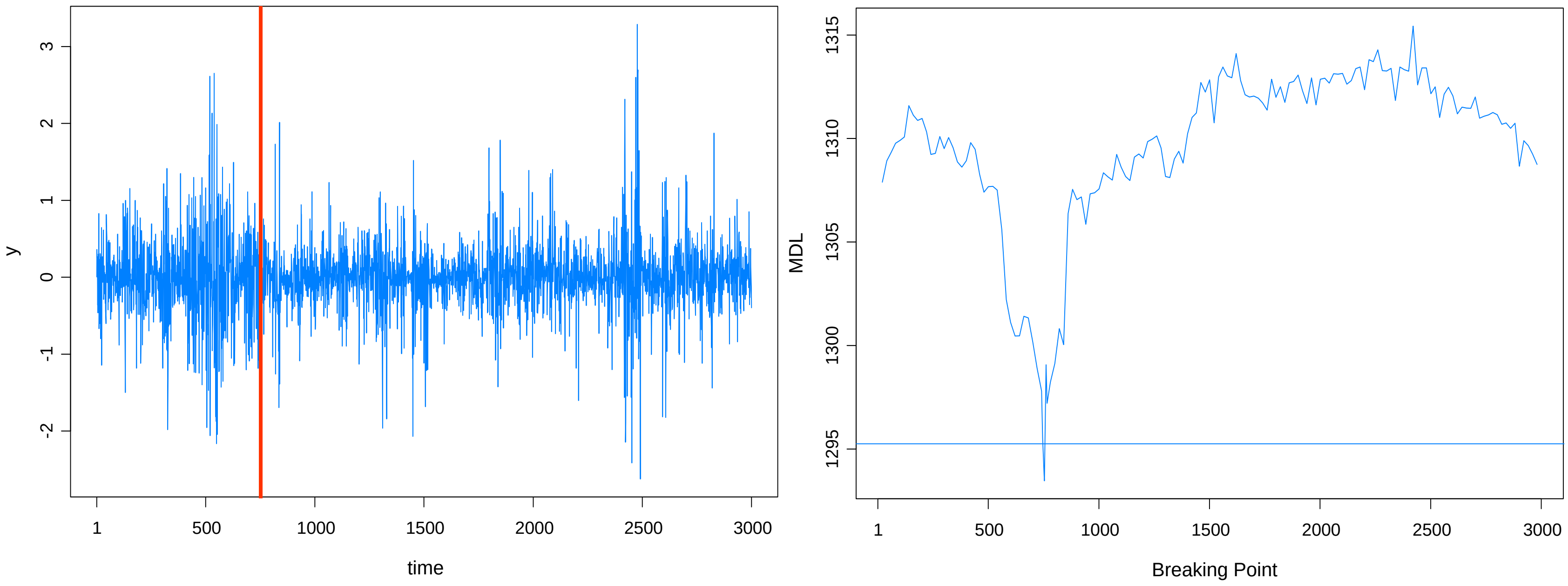
$$Er_a(\psi) = \int \exp\{R(\alpha_n;\alpha^*)\}p_a(\alpha_n\,|\,\mathrm{y}_n;\psi)\,\mathrm{d}\alpha_n.$$

It turns out that $\log(Er_a(\psi))$ is nearly linear and can be approximated by a linear function via importance sampling,

$$e(\psi) \sim e(\hat{\psi}_{AL}) + \dot{e}(\hat{\psi}_{AL})(\psi - \hat{\psi}_{AL})$$

SV Process Example

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

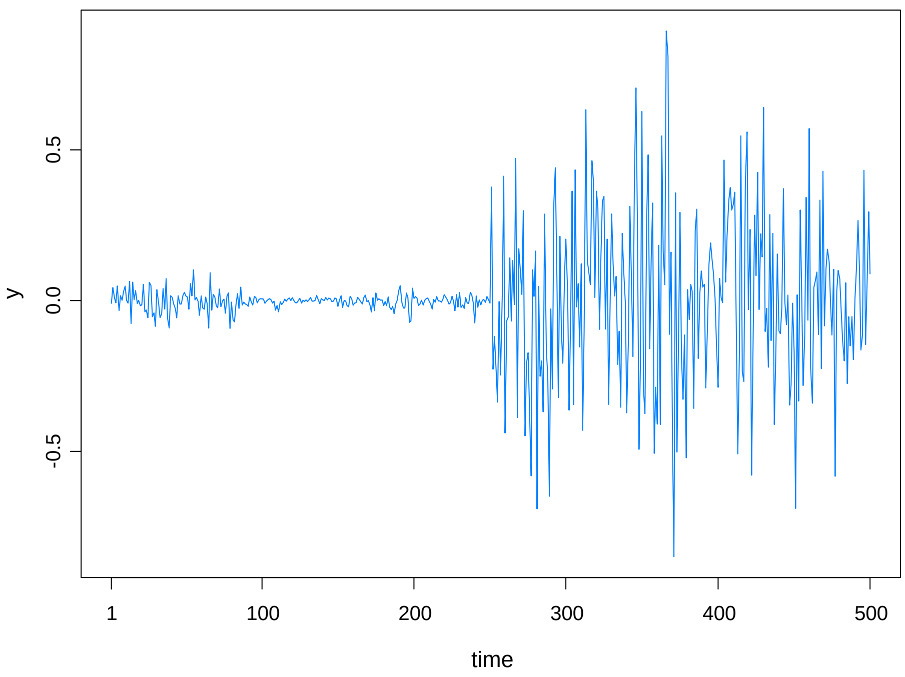


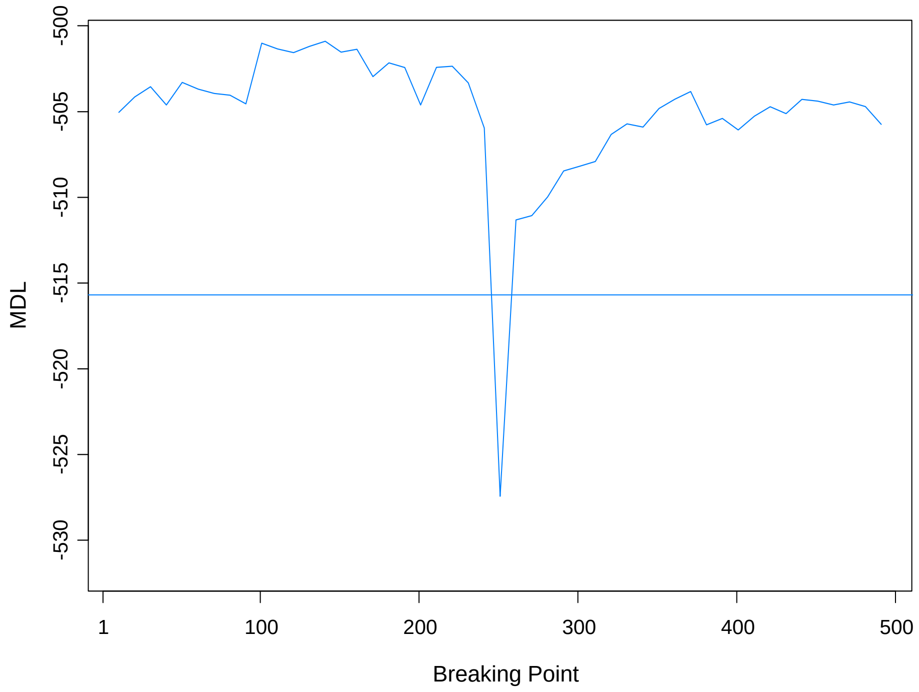
True model:

- $Y_t \mid \alpha_t \sim N(0,\ \exp\{\alpha_t\}),\ \ \alpha_t = -.05 + .975\alpha_{t-1} + \varepsilon_t\ ,\ \ \{\varepsilon_t\}\sim \text{IID N}(0,\ .05),\ \ t \le 750$
- $Y_t \mid \alpha_t \sim N(0,\ \exp\{\alpha_t\ \}),\ \ \alpha_t = -.25 + .900\alpha_{t-1} + \varepsilon_t\ ,\ \ \{\varepsilon_t\}\sim \text{IID N}(0,\ .25),\ \ t > 750.$
- GA estimate 754, time 1053 secs

SV Process Example

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





True model:

- $Y_t \mid \alpha_t \sim N(0,\ \exp\{\alpha_t\}),\ \ \alpha_t = -.175 + .977\alpha_{t-1} + \varepsilon_t\ ,\ \ \{\varepsilon_t\}\sim\text{IID N}(0,\ .1810),\ \ t \le 250$
- $Y_t \mid \alpha_t \sim N(0,\ \exp\{\alpha_t\ \}),\ \ \alpha_t = -.010 + .996\alpha_{t-1} + \varepsilon_t\ ,\ \ \{\varepsilon_t\}\sim\text{IID N}(0,\ .0089),\ \ t > 250.$
- GA estimate 251, time 269s

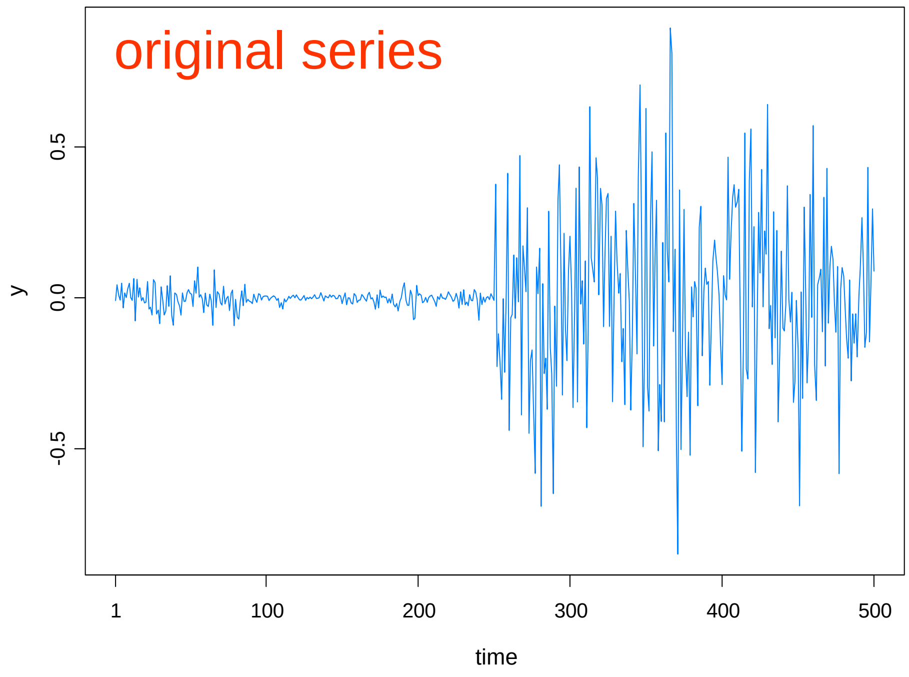
SV Process Example-(cont)

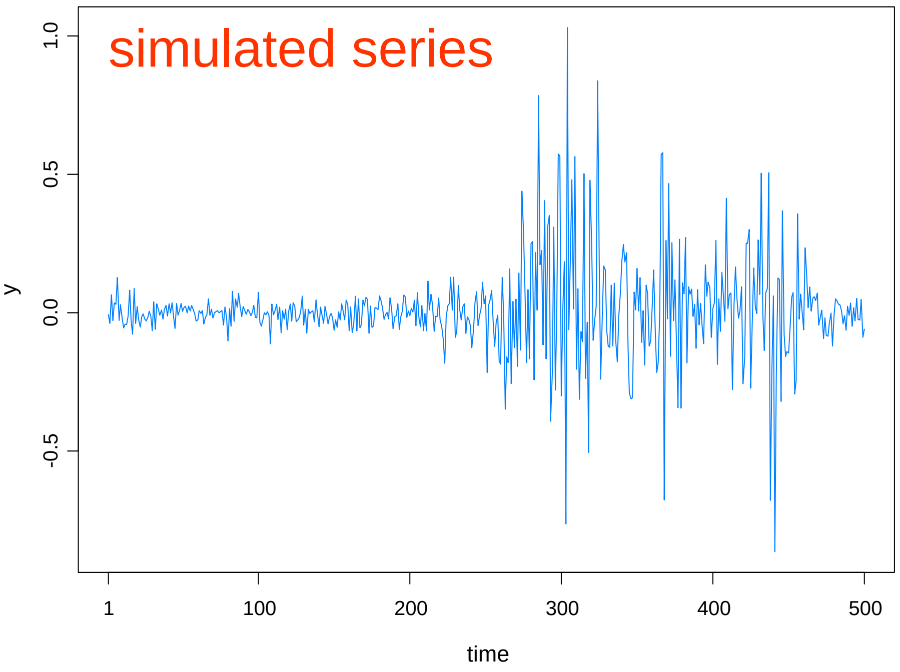
True model:

- $Y_t \mid \alpha_t \sim N(0, \exp\{a_t\})$, $\alpha_t = -.175 + .977\alpha_{t-1} + e_t$, $\{\varepsilon_t\}$~IID N(0, .1810), $t \leq 250$
- $Y_t \mid \alpha_t \sim N(0, \exp\{\alpha_t\})$, $\alpha_t = -.010 + .996\alpha_{t-1} + \varepsilon_t$, $\{\varepsilon_t\}$~IID N(0, .0089), $t > 250$.

Fitted model based on no structural break:

- $Y_t \mid \alpha_t \sim N(0, \exp\{\alpha_t\})$, $\alpha_t = -.0645 + .9889\alpha_{t-1} + \varepsilon_t$, $\{\varepsilon_t\}$~IID N(0, .0935)

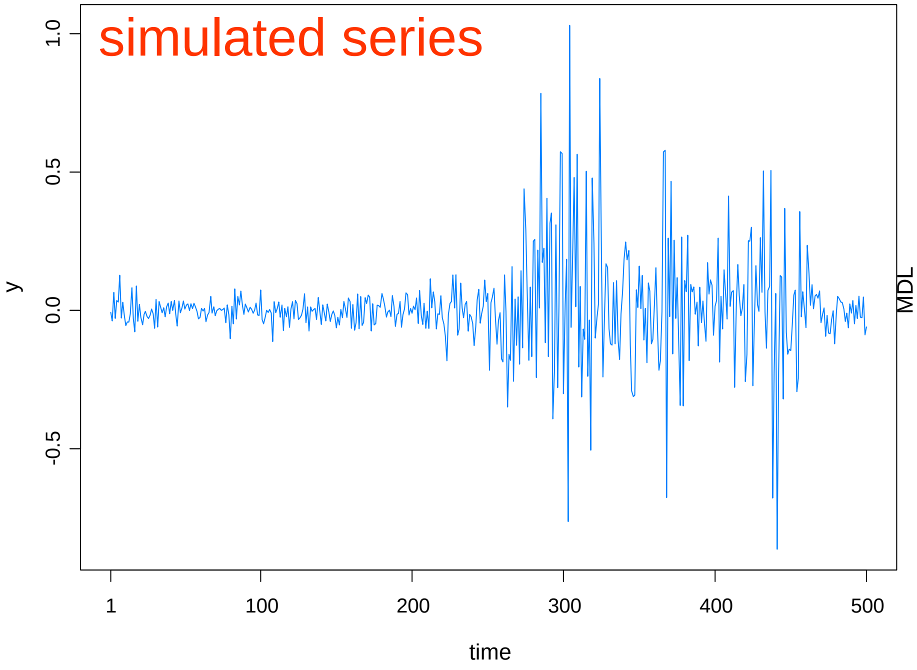


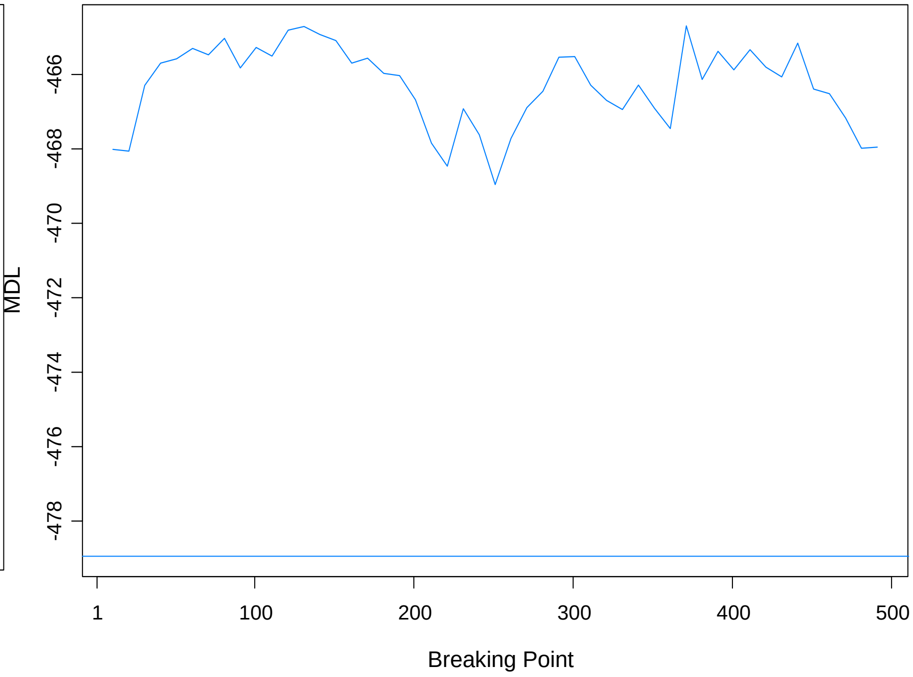


SV Process Example-(cont)

Fitted model based on no structural break:

- $Y_t \mid \alpha_t \sim N(0, \exp\{\alpha_t\})$, $\alpha_t = -.0645 + .9889\alpha_{t-1} + \varepsilon_t$, $\{\varepsilon_t\}$~IID N(0, .0935)





Comments on the theory for AutoPARM

Consistency of $\widehat{m}$:

Suppose the true number of change points is m and let

$$\lambda_1=\tau_1/n, \ . \ . \ . \ , \ \lambda_m=\tau_m/n$$

be the relative (true) change-points. Then AutoPARM estimates of m and λ_j satisfy:

$$\widehat{m} \xrightarrow{p} m,$$

$$\hat{\lambda}_j \xrightarrow{p} \lambda_j.$$

Remarks: The proof is a "cool" application of Strassen's function law of the iterated logarithm.

Comments on the theory for AutoPARM (cont)

Extension: If the process in the jth segment is stationary with ACVF $\gamma_j(h)$ and with $\gamma_j(h) \neq \gamma_{j-1}(h)$ for some lag $h \in \{0,1,\ldots,p^*\}$. Then the estimator which minimizes MDL by fitting AR(p*) models in each segment produces weakly consistent estimators of *m* = true number of change-points.

Remark: This result can be applied to detecting change-points in piecewise GARCH models. Here one can take p*=0 so that we are only talking about changes in variance.

Further extensions: This result has been extended to include a more general framework by Davis and Yau (2012). Base models include ARMA and GARCH—results address some identifiability issues (Andrews and Cheng (2011)).

Comments on the theory for AutoPARM (cont)

Theorem: Suppose $\{X_t\}$ is the AR(1) process

$$X_t = \phi X_{t-1} + \sigma\epsilon_t,$$

where $|\phi| < 1$, and the $\{\epsilon_t\}$ is IID(0,1) with finite 5th moment and pdf f_ϵ satisfying

- $f_\epsilon(x) > 0$ for all x
- $f_\epsilon(x) = f_\epsilon\,(-x)$
- $\mathrm{liminf}_{u\to\infty}\ \mathrm{e}^{\mathrm{cu}} \int_u^\infty f_\epsilon(x)dx) > 0$, for some $c\ > 0$.

Then, MDL is NOT consistent using Yule-Walker estimation. That is, there is a positive probability that a change-point model will be selected as $n \to \infty$.

Modeling Framework for Outlier Detection

Data: $Y_1, \dots, Y_n$

Model parameters

Segments:

m = number of change-points

τ_j = start of jth segment ($\tau_0 = 1 < \tau_1 < . . . < \tau_m < \tau_{m+1} = n$)

Intra-segment parameters (jth segment $\left[\tau_{j-1}, \tau_j\right)$)

Integer valued: $\zeta_j = (p_j, n_j^*, n_j', t_{1,j}^*, \dots, t_{n_j^*,j}^*, t_{1,j}', \dots, t_{n_j',j}')$

- p_j = order of AR process
- A_j = location of additive outliers

  $= \{t_{1,j}^*, \dots, t_{n_j^*,j}^*\}$
- I_j = location of innovational outliers

  $= \{t_{1,j}', \dots, t_{n_j',j}'\}$

Modeling Framework (cont)

Data: $Y_1, \dots, Y_n$

Intra-segment parameters (jth segment $[\tau_{j-1}, \tau_j)$)

Real valued: $\psi_j = (\gamma_j, \phi_{j,1}, \dots, \phi_{j,p_j}, \sigma_j^2, \sigma_j^{*2}, c_j^2)$

- $\left(\gamma_j, \phi_{j,1}, \dots, \phi_{j,p_j}, \sigma_j^2\right)$ = AR model parameters
- $\sigma_j^{*2} \gg \sigma_j^2$ is AO variance
- $c_j^2 \gg 1$ is proportional increase in innovation variance.

Intra-segment AR process with IO:

$$X_{t,j} = \gamma_j + \phi_{j,1} X_{t-1,j} + \cdots + \phi_{j,p_j} X_{t-p_j,j} + \sigma_j \epsilon_{t,j}$$

where $\{\epsilon_{t,j}\}$ is an independent sequence with

$$\text{Var}(\epsilon_{t,j}) = \begin{cases} c_j^2, \text{ if } t \in I_j, \\ 1, \text{ if } t \notin I_j. \end{cases}$$

Modeling Framework (cont)

Data: $Y_1, \dots, Y_n$

Intra-segment AR process with IO:

$$X_{t,j} = \gamma_j + \phi_{j,1} X_{t-1,j} + \cdots + \phi_{j,p_j} X_{t-p_j,j} + \sigma_j \epsilon_{t,j}$$

where $\{\epsilon_{t,j}\}$ is an independent sequence with

$$\text{Var}(\epsilon_{t,j}) = \begin{cases} c_j^2, & \text{if } t \in I_j, \\ 1, & \text{if } t \notin I_j. \end{cases}$$

Intra-segment AR process with IO and AO:

$$Y_t = \begin{cases} X_{t,j} + V_{t,j}, & \text{if } t \in A_j, \\ X_{t,j}, & \text{if t} \notin A_{\text{j}}, \end{cases}$$

where the $\{V_{t,j}\}$ are independent and N(0,σ_j^{*2}) distributed.

Modeling Framework (cont)

Remarks:

- If the $\{\epsilon_{t,j}\}$ and the $\{V_{t,j}\}$ are Gaussian, then model (for fixed integer parameter ζ) can be expressed as a Gaussian linear state-space model.
- Take advantage of state-space model and Kalman recursions for likelihood calculation.

Model Selection Using Minimum Description Length

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).

$\boldsymbol{M}$ = class of operating models for $y = (y_1, \ldots, y_n)$

$L_{\boldsymbol{F}}(y)$ = code length of y relative to $\boldsymbol{F} \in \boldsymbol{M}$

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

$$L_{\boldsymbol{F}}(y) = L(\hat{\boldsymbol{F}}|y) + L(\hat{e}\,|\,\hat{\boldsymbol{F}}),$$

where

$L(\hat{\boldsymbol{F}}|y)$ = code length of the fitted model for $\boldsymbol{F}$

$L(\hat{e}|\hat{\boldsymbol{F}})$ = code length of the residuals based on the fitted model

Model Selection Using Minimum Description Length

Take

M = class of operating models for $y = (y_1, \ldots, y_n)$ that includes piecewise AR models with AO's and IO's.

Let $\eta = (\eta_1, \ldots, \eta_{m+1})$ and $\psi = (\psi_1, \ldots, \psi_{m+1})$ be the integer- and real-valued parameter vectors, where

$$\zeta_j = (p_j, n_j^*, n_j', t_{1,j}^*, \ldots, t_{n_j^*,j}^*, t_{1,j}', \ldots, t_{n_j',j}')$$

$$\psi_j = (\gamma_j, \phi_{j,1}, \ldots, \phi_{j,p_j}, \sigma_j^2, \sigma_j^{*2}, c_j^2).$$

Then

$$\mathrm{L}(\hat{F}|y) = \log m + (m+1)\log n$$

$$+\sum_{j=1}^{m+1} (\log p_j + \log n_j^* + \log n_j' + (n_j^* + n_j')\log n_j + 1/2(p_j + 3)\log n_j)$$

Model Selection Using Minimum Description Length

Moreover, using results from information theory, Rissanen argues

$$\mathrm{L}(\hat{e}|\widehat{F}) \sim -\sum_{j=1}^{m+1} \log L_j(\hat{\psi}_j;\ \boldsymbol{y}_j)$$

where $L_j(\hat{\psi}_j;\ \boldsymbol{y}_j)$ is the likelihood evaluated at the MLE of the j$^{\text{th}}$ segment; so that

$$MDL(m, \tau, p, n', n^*, I, A)$$

$$= \log m + (m+1)\log n + \sum_{j=1}^{m+1} (\log p_j + \log n_j^* + \log n_j'$$

$$+ \left(n_j^* + n_j'\right) \log n_j + 1/2\left(p_j + 3\right) \log n_j) + \sum_{j=1}^{m} (\log_2(2\pi\hat{\sigma}_j^2) + n_j)$$

Model Selection Using Minimum Description Length

Computing the likelihood $L_j(\psi_j;\ \boldsymbol{y}_j)$: Start by computing likelihood in a segment given p and locations of additive and innovational outliers.

$$
\begin{aligned}
L(\psi; \boldsymbol{y}|A, I) &= \int_{\mathbb{R}^n} L(\psi; \boldsymbol{y}, \boldsymbol{x}_A|I)d\boldsymbol{x}_A \\
&= \int_{\mathbb{R}^n} p(\boldsymbol{y}_A, \boldsymbol{y}_{A^c}, \boldsymbol{x}_A|I)d\boldsymbol{x}_A \\
&= \int_{\mathbb{R}^n} p(\boldsymbol{y}_A, \boldsymbol{x}_{A^c}, \boldsymbol{x}_A|I)d\boldsymbol{x}_A \\
&= \int_{\mathbb{R}^n} p(\boldsymbol{y}_A|\boldsymbol{x}_A)p(\boldsymbol{x}\ |I)d\boldsymbol{x}_A
\end{aligned}
$$

Model Selection Using Minimum Description Length

Computing the likelihood $L_j(\psi_j;\ \boldsymbol{y}_j)$:

$$L(\psi; \boldsymbol{y}|A, I) = \int_{\mathbb{R}^n} p(\boldsymbol{y}_A|\boldsymbol{x}_A)p(\boldsymbol{x}\,|I)d\boldsymbol{x}_A$$

- $p(\boldsymbol{y}_A|\boldsymbol{x}_A) \sim\phi(\boldsymbol{y}_A; \boldsymbol{x}_A, \sigma_0^2 I_n)$ multivariate normal

- $p(\boldsymbol{x}|I) =$

$$\phi(x_1, \dots, x_p; \mu, V_0^{-1}) \prod_{\substack{t>p \\ t\notin I}} (2\pi\sigma^2)^{-\frac{1}{2}} \exp\{-\frac{(x_t-\gamma-\phi_1 x_{t-1}-\cdots\phi_p x_{t-p})^2\}}{2\sigma^2}\}$$

$$\times \prod_{\substack{t>p \\ t\in I}} (2\pi c^2\sigma^2)^{-\frac{1}{2}} \exp\{-\frac{(x_t-\gamma-\phi_1 x_{t-1}-\cdots\phi_p x_{t-p})^2\}}{2c^2\sigma^2}\}$$

Remark: Use Kalman recursions to compute likelihood.

Optimization Using Genetic Algorithm

Genetic Algorithm: Chromosome consists of *n* genes, each taking the value of -3 (AO), -2 (AI), –1 (no break), *p* (order of AR process).

Use natural selection to find a *near* optimal solution.

Map the break points with a chromosome *c* via

$(m, \tau, p, A, I) \leftrightarrow c = (\delta_1, \ldots, \delta_n)$ where

$$\delta_t = \begin{cases} -3, & \text{if AO occurs at } t, \\ -2, & \text{if IO occurs at } t, \\ -1, & \text{if no break occurs at } t, \\ p, & \text{if break occurs at time } t \text{ and AR model is } p. \end{cases}$$

For example,

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

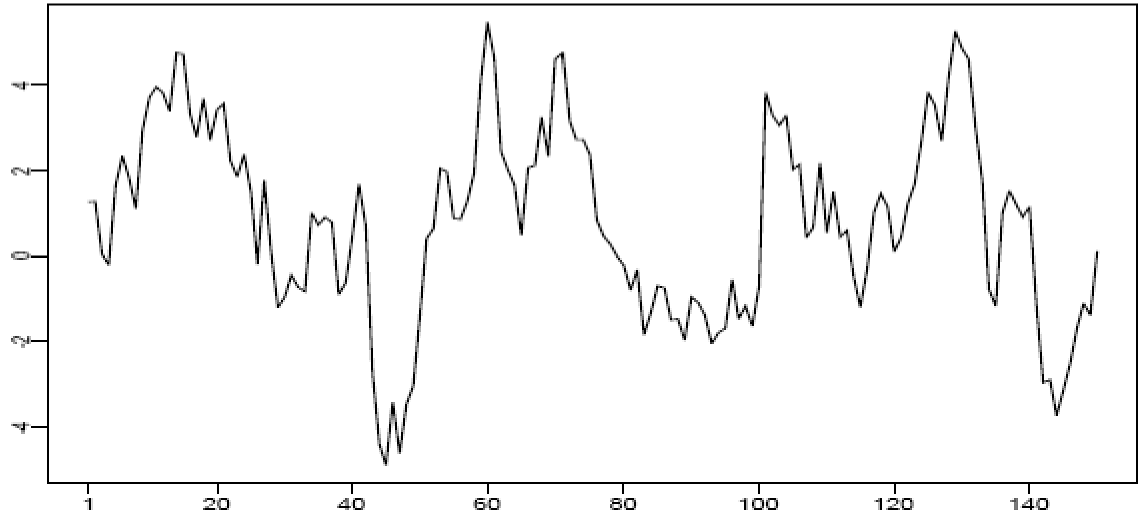
t: 1 6 11 15

would correspond to a process with AO at t=3, IO at t=18.

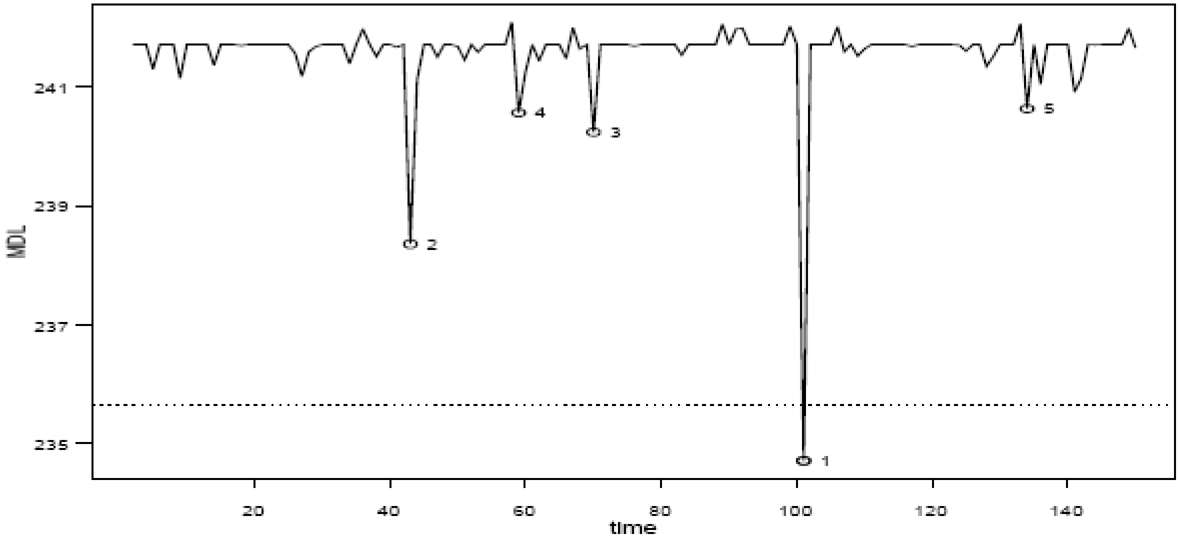
AR(2), t=1:5; AR(0), t=6:10; AR(0), t=11:14; AR(3), t=15:20

Robust AutoPARM-IO contaminated AR model

Single AR model with single IO at ?

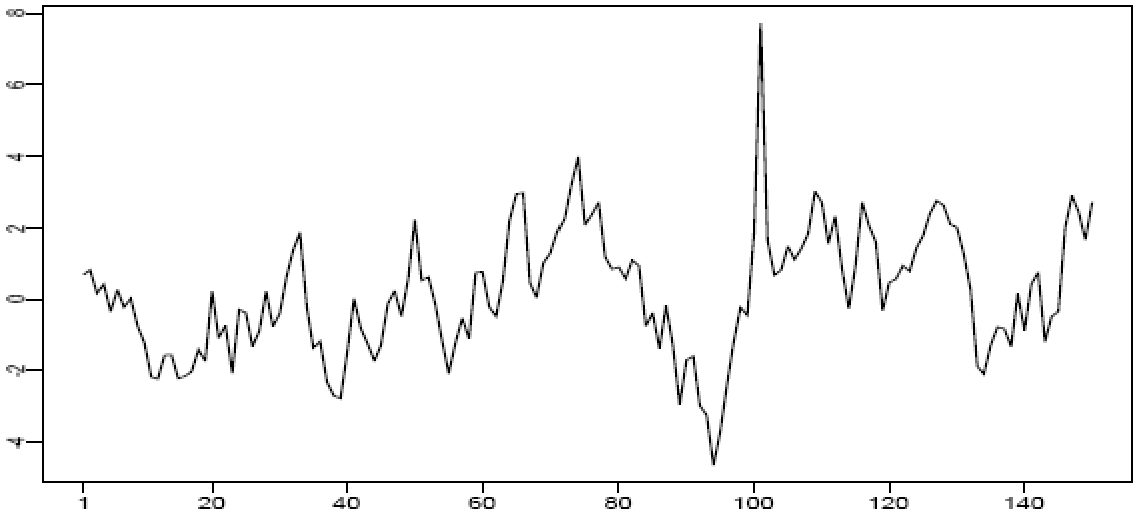


MDL’s w/ top 5 residuals indicated

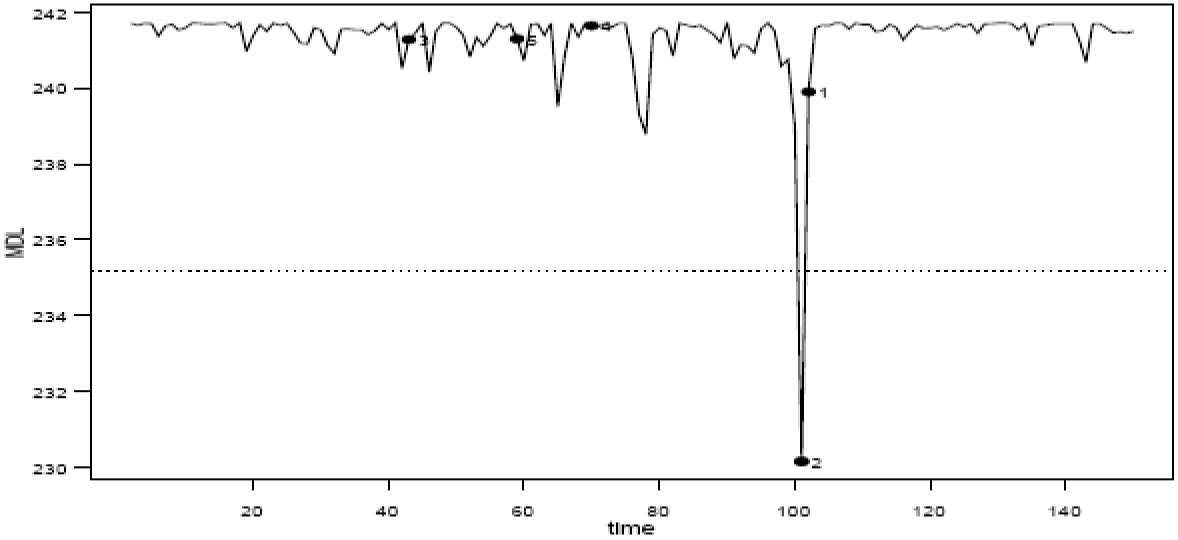


Robust AutoPARM-AO contaminated AR model

Single AR model with single AO at 101



MDL's w/ top 5 residuals indicated



Accelerating Robust AutoPARM

Remark: Observe that large reductions in MDL due to ***outliers*** seem to occur at locations of largest residuals. Use this info to facilitate the search optimizer.

Search modification: for simplicity, assume one AR(p) signal

1. Select locations $S_1 = \{t_1, \ldots, t_k\}$ of *k* largest absolute residuals.
2. Let $M^a(t)$ (resp $M^i(t)$) be an AR(p) model with AO(IO) at time ***t*** and set

$$F_1 = \{M^a(t), M^i(t),\ \ t \in S_1\}$$

3. Define $M_1 = \mathrm{argmin}_{\mathrm{F} \in F_1} MDL(F);\, r_1 =$ location of outlier
4. Set $F_2 = \{M^a(r_1, t), M^i(r_1, t),\ \ t \in S_1 - \{r_1\}\}$

$M_2 = \mathrm{argmin}_{\mathrm{F} \in F_2} MDL(F);\, r_2 =$ location of outlier

5. Continue to obtain $M_1, \ldots, M_k$ and define

$$q = \arg\min_{0 \le j \le k} MDL(M_j)$$

Accelerating Robust AutoPARM

6. Only allow time locations $r_1, \dots, r_q$ for outliers in the current generation of the genetic algorithm.

7. After producing the next generation, select largest residuals again and repeat.

Remark: The outlier selection step has elements that are similar in spirit to one proposed by Bianco et al. (2001) and Sánchez and Peña (2003). Our method is not as likely to stop prematurely.

Accelerating Robust AutoPARM

Example: AR(1) model

$$X_t = .8X_{t-1} + \epsilon_t, \qquad \{\epsilon_t\} \sim IID\ N(0,1)$$

with AO's at {16,32,48,80} and no IOs.

Residuals selected by minimizing MDL

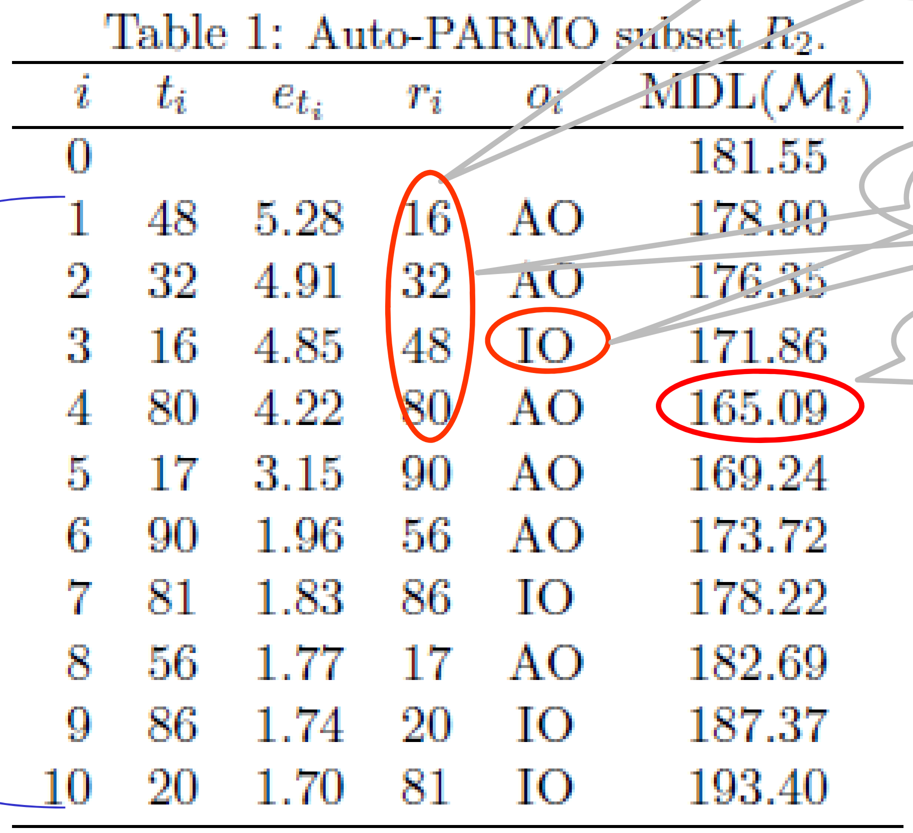


Table 1: Auto-PARMO subset R_2.

i	t_i	e_{t_i}	r_i	o_i	MDL($\mathcal{M}_i$)
0					181.55
1	48	5.28	16	AO	178.90
2	32	4.91	32	AO	176.35
3	16	4.85	48	IO	171.86
4	80	4.22	80	AO	165.09
5	17	3.15	90	AO	169.24
6	90	1.96	56	AO	173.72
7	81	1.83	86	IO	178.22
8	56	1.77	17	AO	182.69
9	86	1.74	20	IO	187.37
10	20	1.70	81	IO	193.40

Correctly identified locations

One wrong classification

Minimum MDL

10 largest residuals

Robust AutoPARM-sample results

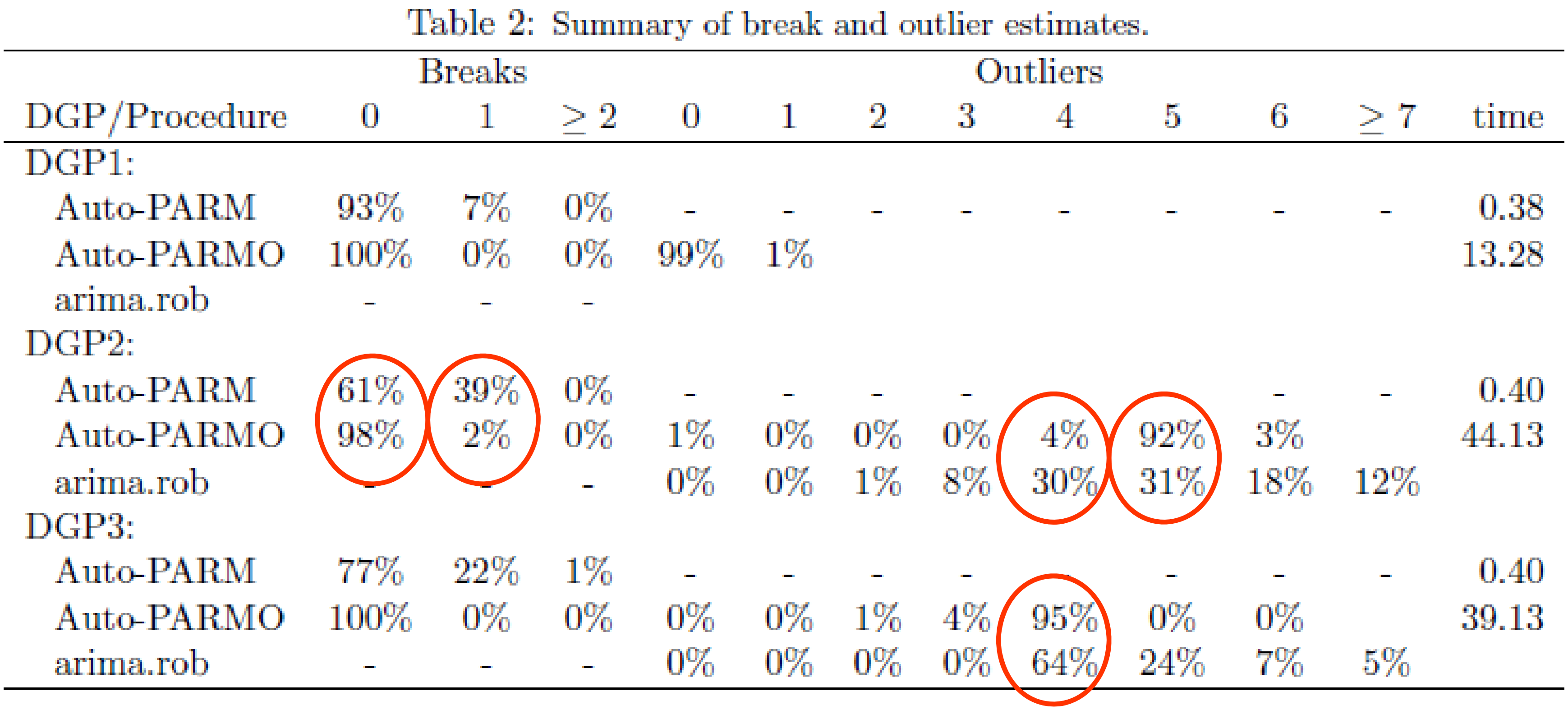


Table 2: Summary of break and outlier estimates.

	Breaks			Outliers								
DGP/Procedure	0	1	≥ 2	0	1	2	3	4	5	6	≥ 7	time
DGP1:												
Auto-PARM	93%	7%	0%	-	-	-	-	-	-	-	-	0.38
Auto-PARMO	100%	0%	0%	99%	1%							13.28
arima.rob	-	-	-									
DGP2:												
Auto-PARM	61%	39%	0%	-	-	-	-			-	-	0.40
Auto-PARMO	98%	2%	0%	1%	0%	0%	0%	4%	92%	3%		44.13
arima.rob	-	-	-	0%	0%	1%	8%	30%	31%	18%	12%	
DGP3:												
Auto-PARM	77%	22%	1%	-	-	-	-	-	-	-	-	0.40
Auto-PARMO	100%	0%	0%	0%	0%	1%	4%	95%	0%	0%		39.13
arima.rob	-	-	-	0%	0%	0%	0%	64%	24%	7%	5%	

DGP1: one-piece AR model; no outliers

DGP2: one-piece AR model; 5 innovation outliers

DGP3: one-piece AR model; 4 additive outliers

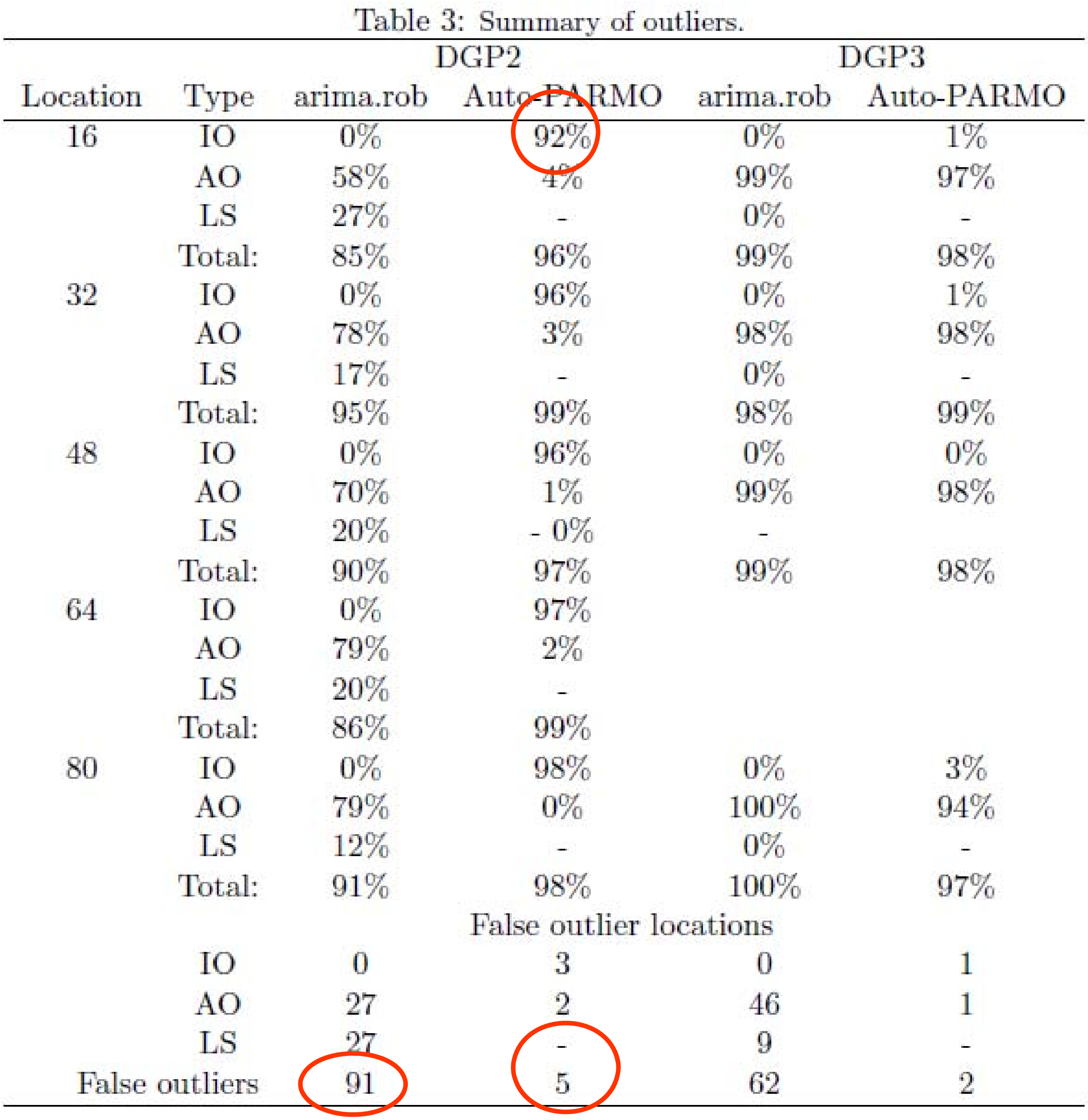


Table 3: Summary of outliers.

		DGP2		DGP3	
Location	Type	arima.rob	Auto-PARMO	arima.rob	Auto-PARMO
16	IO	0%	92%	0%	1%
	AO	58%	4%	99%	97%
	LS	27%	-	0%	-
	Total:	85%	96%	99%	98%
32	IO	0%	96%	0%	1%
	AO	78%	3%	98%	98%
	LS	17%	-	0%	-
	Total:	95%	99%	98%	99%
48	IO	0%	96%	0%	0%
	AO	70%	1%	99%	98%
	LS	20%	- 0%	-	
	Total:	90%	97%	99%	98%
64	IO	0%	97%		
	AO	79%	2%		
	LS	20%	-		
	Total:	86%	99%		
80	IO	0%	98%	0%	3%
	AO	79%	0%	100%	94%
	LS	12%	-	0%	-
	Total:	91%	98%	100%	97%
		False outlier locations			
	IO	0	3	0	1
	AO	27	2	46	1
	LS	27	-	9	-
False outliers		91	5	62	2

Robust AutoPARM-sample results

Three-piece AR model:

$$Y_t = \begin{cases} .8Y_{t-1} + 3\epsilon_t, & \text{if } 1 < t \leq 128, \\ 1.69Y_{t-1} - .81Y_{t-2} + \epsilon_t, & \text{if } 128 < \mathrm{t} \leq 192, \\ 1.32Y_{t-1} - .81Y_{t-2} + 3\epsilon_t, & \text{if } 192 < \mathrm{t} \leq 256. \end{cases}$$

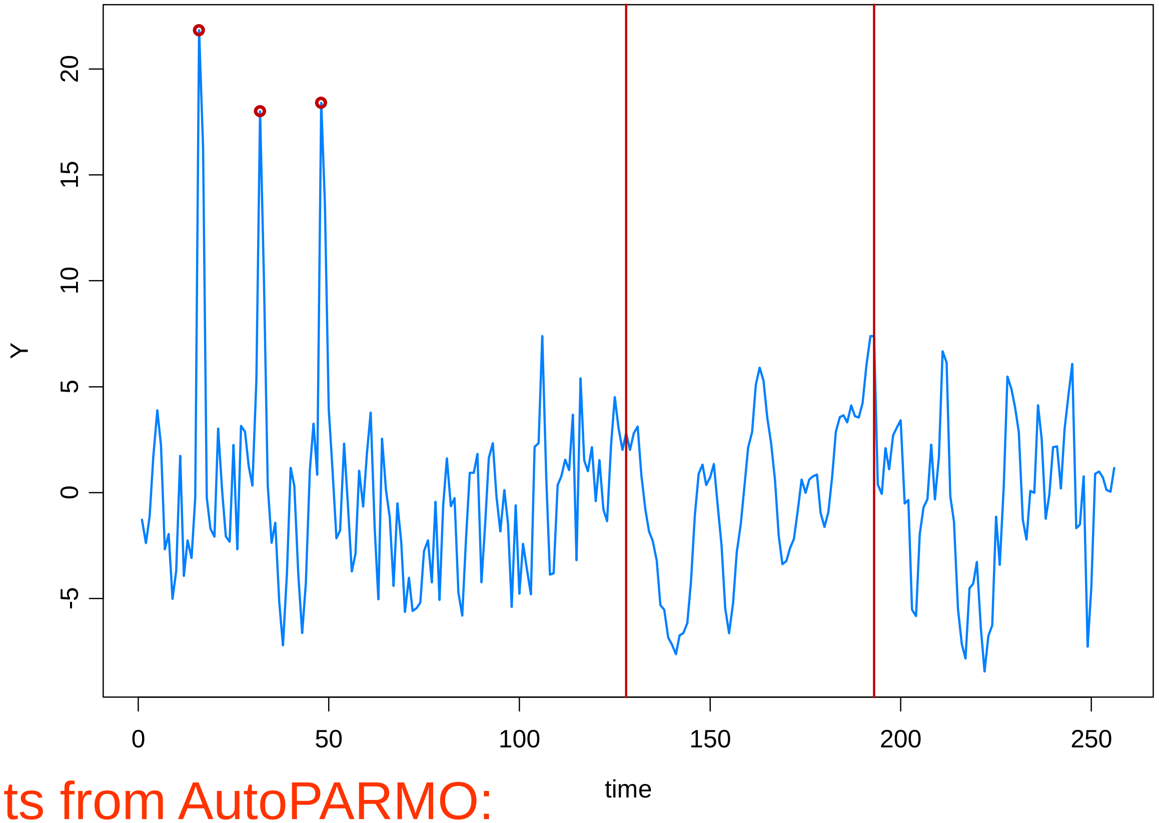
DGP4: three-piece AR model; no outliers

DGP5: three-piece AR model; 3 innovation outliers at 16, 32, 48.

DGP6: three-piece AR model; 3 additive outliers at 16, 32, 48.

Robust AutoPARM-sample results

DGP5: three-piece AR model; 3 innovation outliers at 16, 32, 48.



Results from AutoPARMO:

Breaks: 124 and 194; 3 IA outliers: 16, 32, 48

Results from AutoPARM(no outliers): Breaks at **52**, 126, 294

arima.rob: AO at 16, 17,32, 48, 49

Robust AutoPARM-sample results

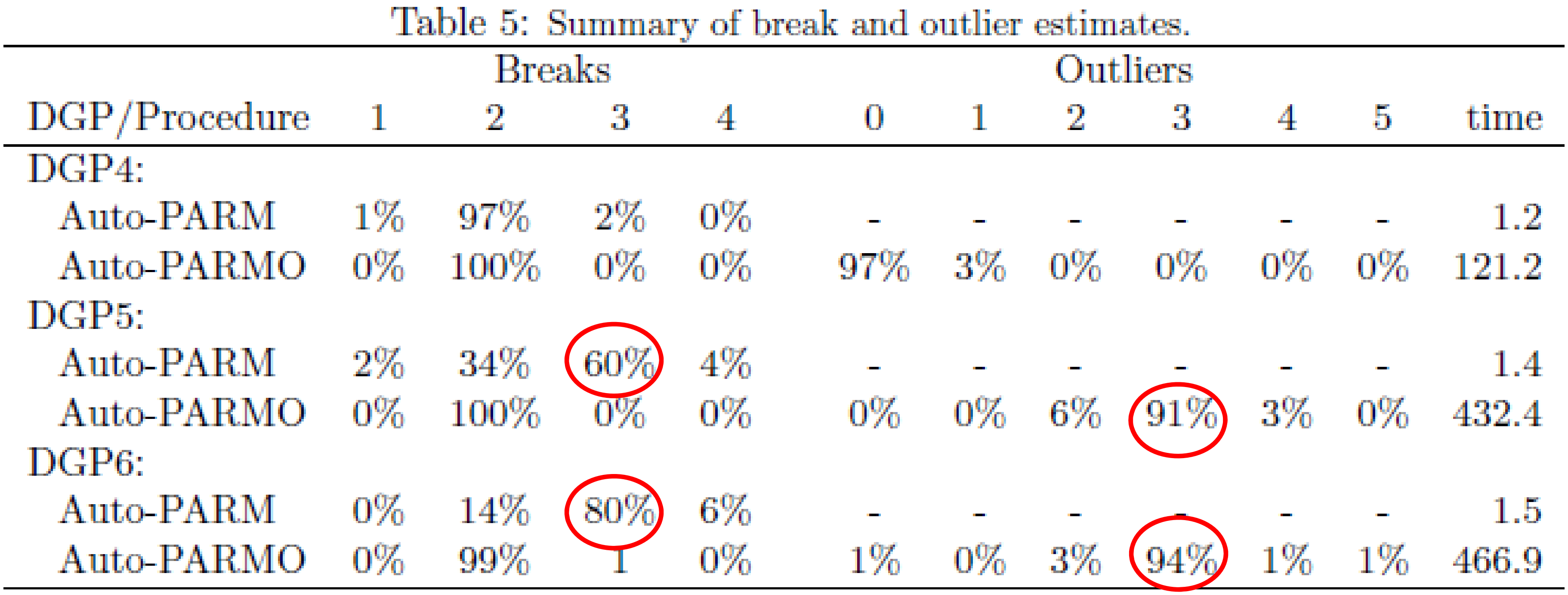


Table 5: Summary of break and outlier estimates.

	Breaks				Outliers						
DGP/Procedure	1	2	3	4	0	1	2	3	4	5	time
DGP4:											
Auto-PARM	1%	97%	2%	0%	-	-	-	-	-	-	1.2
Auto-PARMO	0%	100%	0%	0%	97%	3%	0%	0%	0%	0%	121.2
DGP5:											
Auto-PARM	2%	34%	60%	4%	-	-	-	-	-	-	1.4
Auto-PARMO	0%	100%	0%	0%	0%	0%	6%	91%	3%	0%	432.4
DGP6:											
Auto-PARM	0%	14%	80%	6%	-	-	-	-	-	-	1.5
Auto-PARMO	0%	99%	1	0%	1%	0%	3%	94%	1%	1%	466.9

DGP4: three-piece AR model; no outliers

DGP5: three-piece AR model; 3 innovation outliers

DGP6: three-piece AR model; 3 additive outliers

Robust AutoPARM-sample results

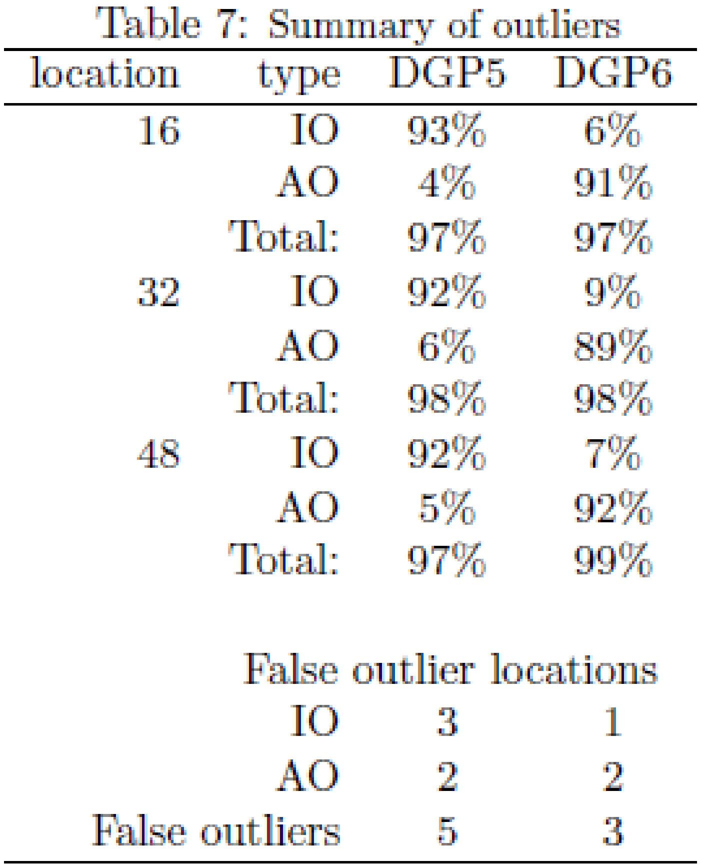


Table 7: Summary of outliers

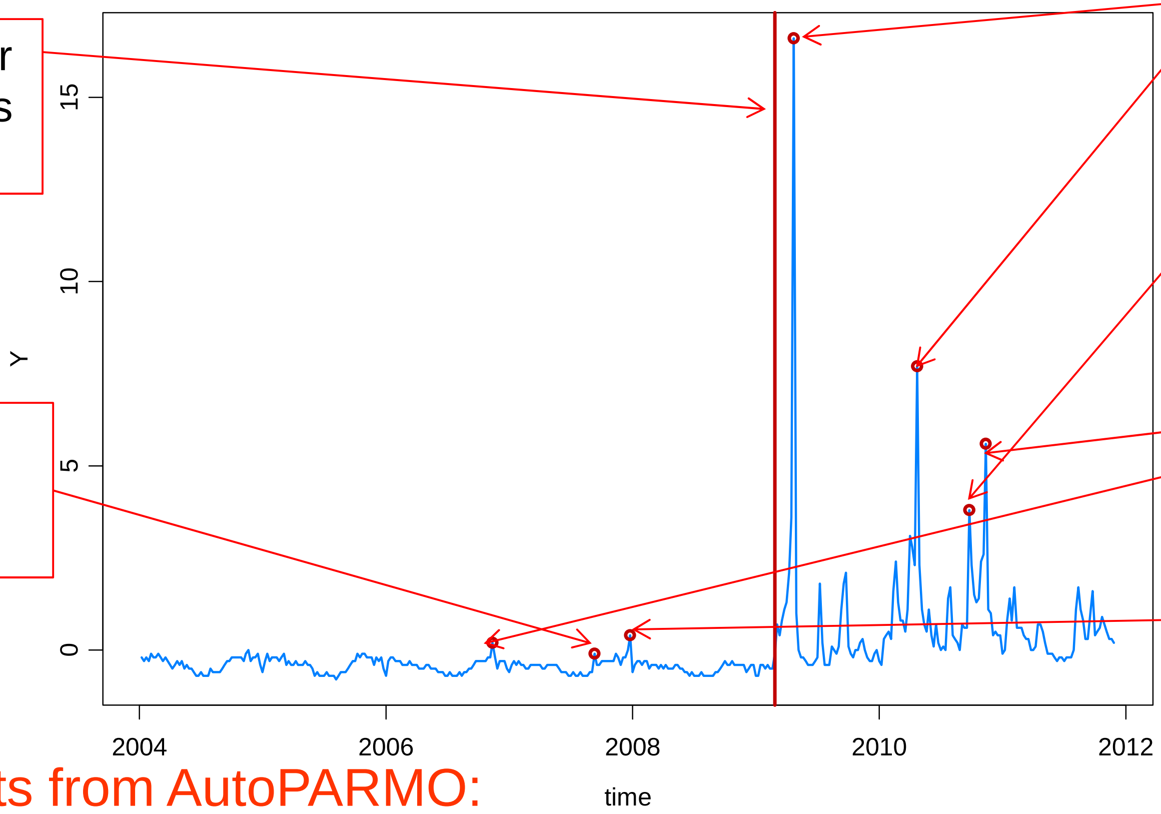
location	type	DGP5	DGP6
16	IO	93%	6%
	AO	4%	91%
	Total:	97%	97%
32	IO	92%	9%
	AO	6%	89%
	Total:	98%	98%
48	IO	92%	7%
	AO	5%	92%
	Total:	97%	99%
	False outlier locations		
	IO	3	1
	AO	2	2
	False outliers	5	3

DGP5: two-piece AR model; 3 innovation outliers

DGP6: two-piece AR model; 3 additive outliers

Application to Google Trends Example: Tea Party

Weekly data: January 2004--December 2011.



3 weeks after Obama takes office

Texas Tea Party straw poll

Taxes due

9-12 taxpayer march on Washington DC

Midterm elections

Ron Paul raised $4.4M on internet in 1 day

Results from AutoPARMO:

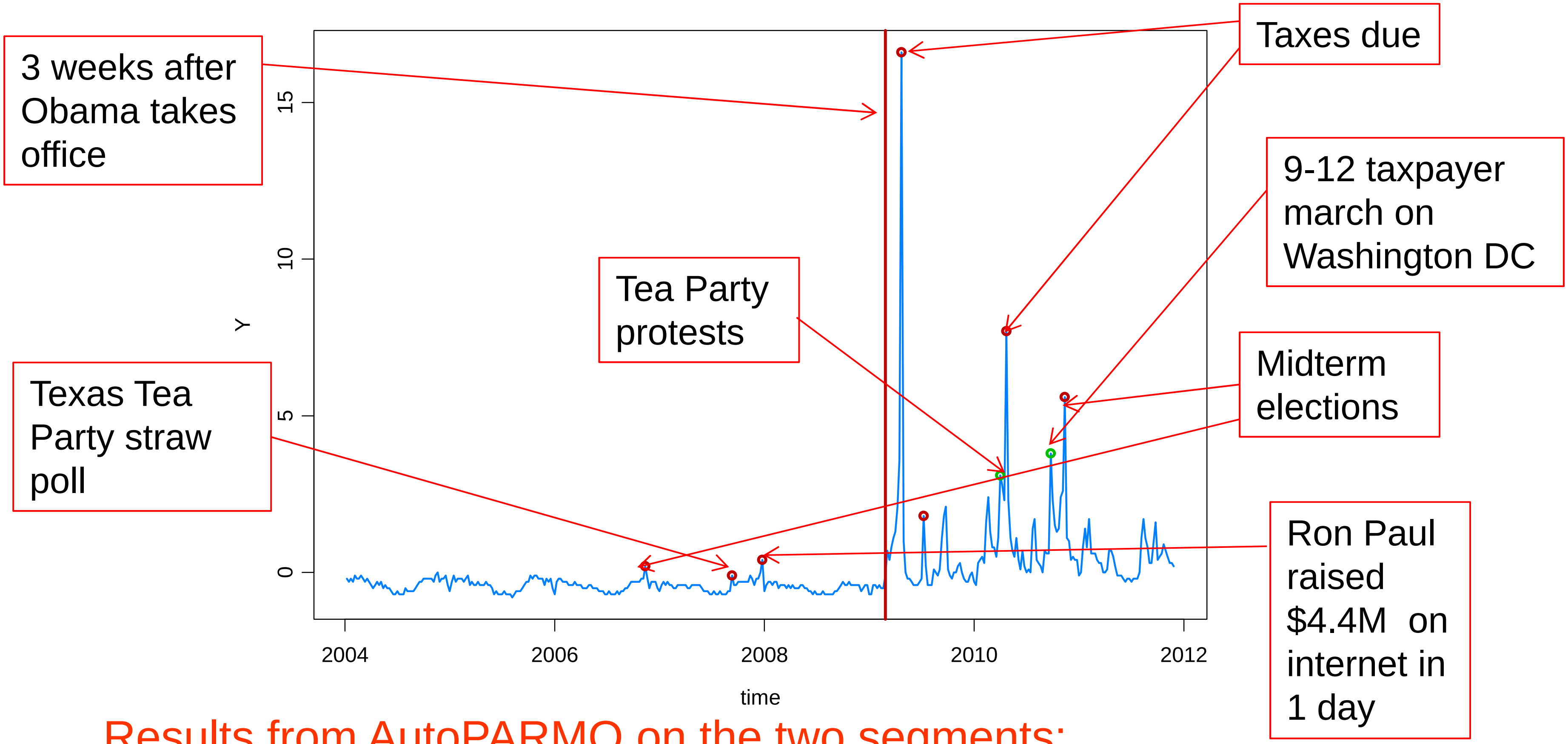
Break: Feb 15, 2009 (3 weeks after Obama took office)

AO outliers: 11-5-06(election week); 9-2-07; 12-16-07;

4-12-09 (taxes due); 4-11-10 (taxes); 9-12-10; 10-31-10 (election)

Application to Google Trends Example: Tea Party

Refinement: January 2004--December 2011.



3 weeks after Obama takes office

Taxes due

9-12 taxpayer march on Washington DC

Tea Party protests

Midterm elections

Texas Tea Party straw poll

Ron Paul raised $4.4M on internet in 1 day

Results from AutoPARMO on the two segments:

AO outliers (7): 149, 192, 207, 276, 287, 328, 357

IO outliers (2): 325, 350

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

4. Extensions to *outlier* (both innovation and additive) *detection* are currently under study. Results look promising—new implementation is almost ready.

Running the Program AutoPARM

- Download the zip file **AutoPARM** (only runs on PC) from
  www.stat.columbia.edu/~rdavis/AutoPARM/AutoPARM.zip
- Extract the zip file (folder called AutoPARM) will be created.
- Open a command prompt window
  (In run, type the command cmd)
- Navigate to the AutoPARM folder
- AutoPARM needs an input file (first several lines of example file eqn5.in are:
  - DATA:
    eqn5.dat [name of file containing input data]
  - OUTPUT:
    eqn5.out [name of file to export results]
- Execute program by typing on command line

  ```
  AutoPARM eqn5.in
  ```

- Results are written to file named eqn5.out
- More details about running the program can be found in AutoPARM.pdf