Thoughts on Model Selection

Application of MDL to Change Point Estimation

Richard A. Davis
Colorado State University

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

Gabriel Rodriguez-Yam, CSU

Thomas Lee, CSU

PROGRAM

- Akaike Information Criterion
  - Kullback-Leibler
  - Adjustments
  - Muddy Creek example
  - Other criteria
  - Consistency/efficiency
- Model Selection Using Minimum Description Length (MDL)
  - Fitting piecewise AR models
  - Optimization using the genetic algorithm
- Examples
  - simulation
  - real
- Application to nonlinear models

Akaike Information Criterion (AIC)

References on AIC: Linhart and Zucchini (1986), Burnham and Anderson (1998) and McQuarrie and Tsai (1998).

Data: $\mathrm{Y} = (Y_1, \ldots, Y_n)$

Models: ***M*** = family of operating models for Y.

Sometimes referred to as candidate (statistical) models for Y

AIC: for a given model $F \in M$ define

$$\mathrm{AIC}(F) = -2 \log L(\hat{F}|y) + 2p,$$

where

- $L(\hat{F}|y)$ is the likelihood evaluated at the MLE for model ***F.***
- p = parameter dimension for model ***F.***

Goal: choose model $F \in M$ that *minimizes* AIC(***F***).

AIC (cont)

Example: Order selection for autoregressive (AR) time series models.

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \cdots + \phi_p Y_{t-p} + \varepsilon_t,$$

where $\{\varepsilon_t\}$ is IID $N(0,\sigma^2)$.

Model: ***M*** = class of operating models = AR(p) models, $p \geq 0$.

AIC:

$$\text{AIC}(p = -2 \log L(\hat{\phi}, \hat{\sigma}^2 | y) + 2p,$$

Goal: choose p that *minimizes* AIC(p).

AIC (cont)

Akaike's idea: AIC is intended to be an *approximately unbiased* estimator of the *Kullback-Leibler discrepancy* between the candidate and true models.

K-L discrepancy:

$$I(\boldsymbol{F}) = \int -2\log\left(\frac{f(y\,|\,\boldsymbol{F})}{f_T(y)}\right) f_T(y)\,dy$$

$$\geq 0$$

with equality if and only if $f(.\,;\boldsymbol{F}) = f_T$. K-L discrepancy is a measure of the

- distance between $f(.\,;\boldsymbol{F})$ and f_T
- loss of information when $f(.\,;\boldsymbol{F})$ is used as the model instead of f_T

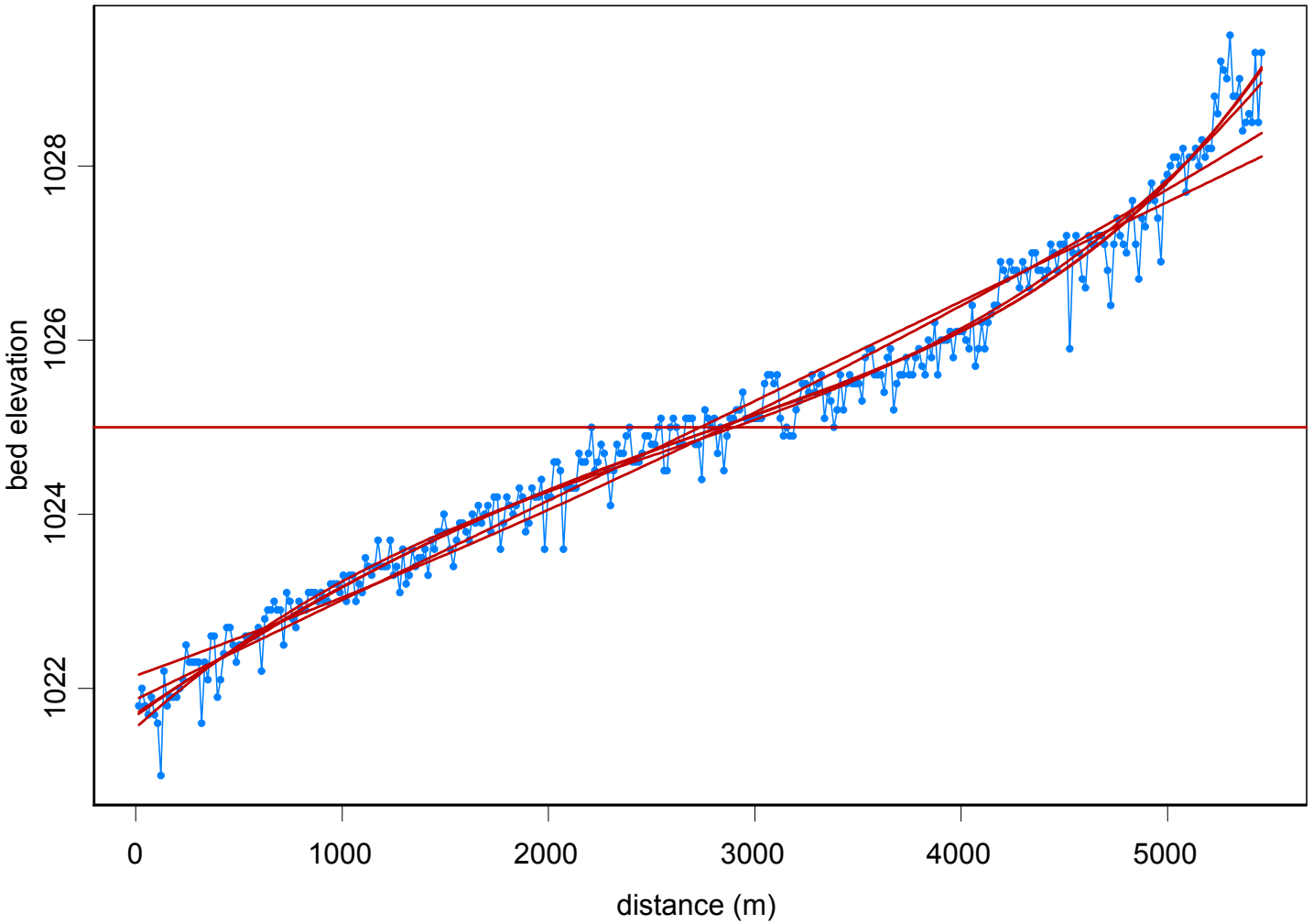
Adjustments:

- $2p$ or not $2p$
- small sample correction (Sugiuraa `78; Hurvich and Tsai `89)

$$\mathrm{AICC}(p) = -2\log L(\hat{\phi}|y) + 2(p+1)n/(n-p-2),$$

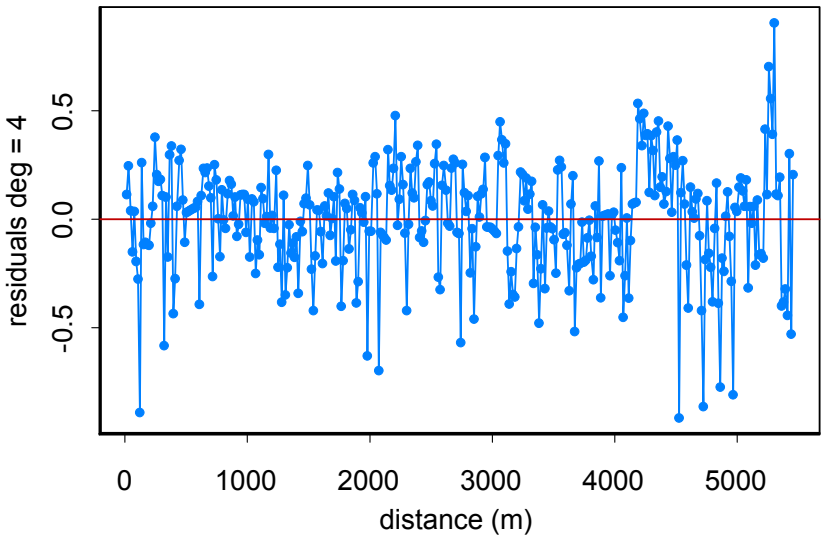
Muddy Creek- tributary to Sun River in Central Montana

Muddy Creek: surveyed every 15.24 meters, total of 5456m; 358 measurements



Degree	AIC_c
0	1455
1	294.3
2	251.3
3	47.1
4	34.0
5	35.5

Muddy Creek: residuals from poly(d=4) fit

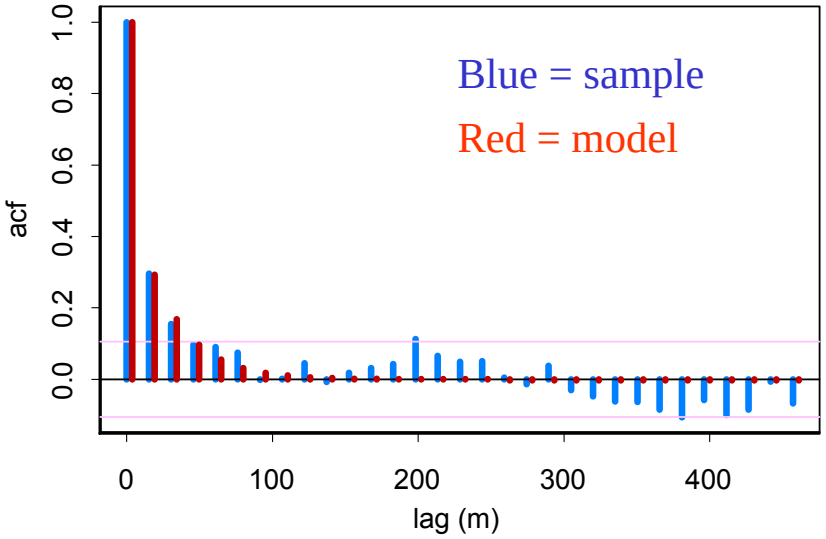


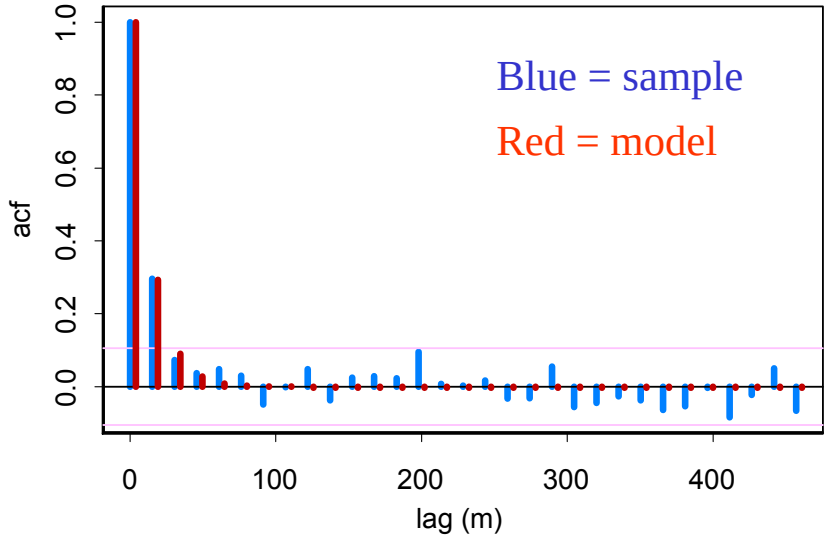
Minimum AIC_c ARMA model: ARMA(1,1)

$Y_t = .574\ Y_{t-1} + \varepsilon_t - .311\ \varepsilon_{t-1}$, $\{\varepsilon_t\}$~WN(0,.0564)

Noncausal ARMA(1,1) model:

$Y_t = 1.743\ Y_{t-1} + \varepsilon_t - .311\ \varepsilon_{t-1}$





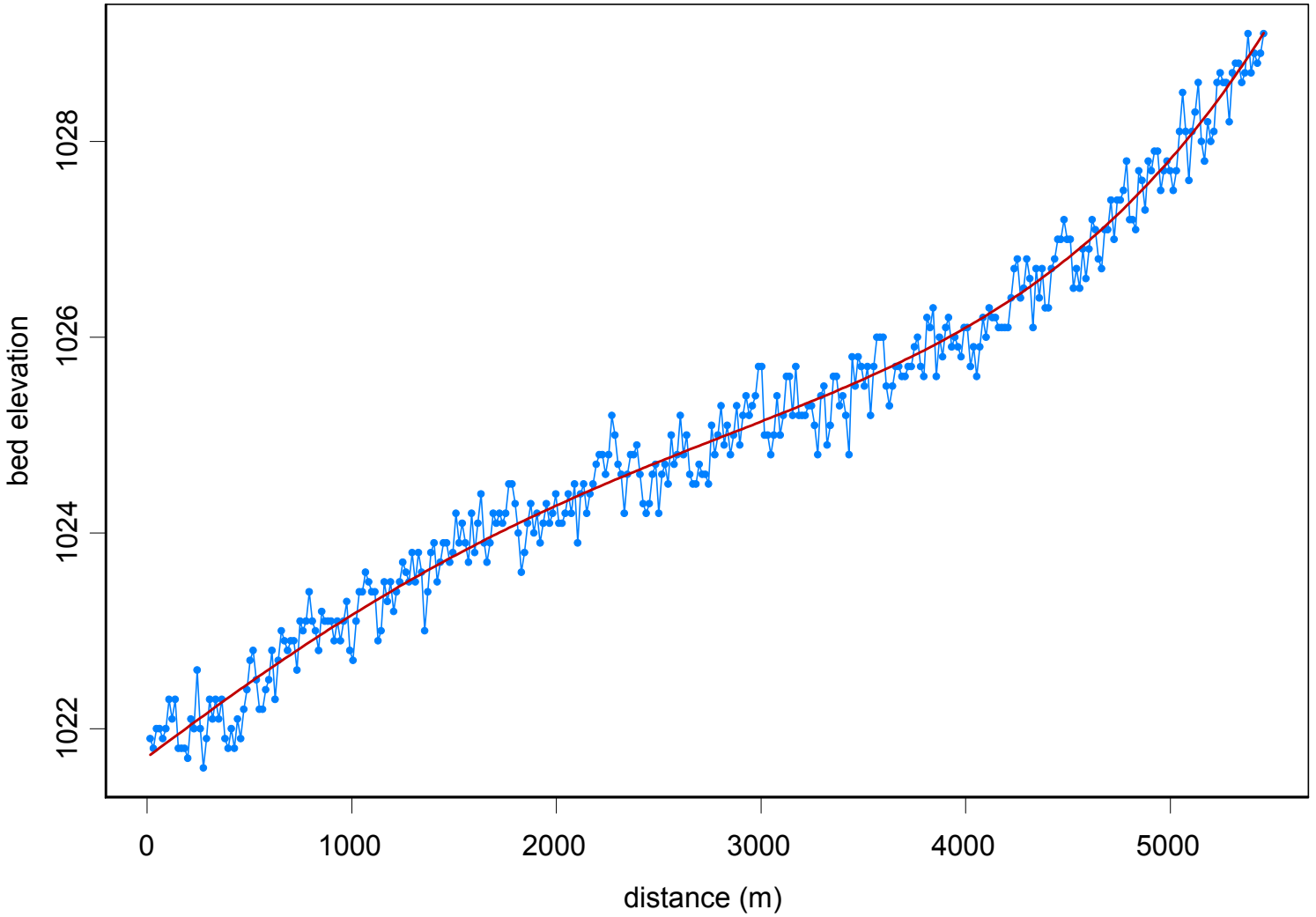
Muddy Creek (cont)

Summary of models fitted to Muddy Creek bed elevation:

Degree	AIC_c	ARMA	AIC_c
0	1455	(1,2)	59.67
1	294.3	(2,1)	26.98
2	251.3	(2,1)	26.30
3	47.1	(1,1)	7.12
4	**34.0**	**(1,1)**	**2.78**
5	35.5	(1,1)	4.68

Muddy Creek (cont)

Simulated series: polynomial degree 4 + ARMA(1,1):



A Brief History (of AIC) in Time

Akaike `73: developed AIC

Hannan and Quinn `79: showed that AIC was *NOT* consistent. That is the AIC estimate of p did not necessarily converge to the true AR order.

Hannan and Quinn `79 and Hannan `80: showed that BIC was consistent. That is the BIC estimate of p converges to the true AR order.

Shibata `80: showed AIC was efficient (for prediction). Here true model is an infinite order AR process.

Hurvich and Tsai `89: argued for a return to AIC based on K-L discrepancy considerations.

Burnham and Anderson `98: popularized the use of AIC in ecology.

Other criteria

AICC:

$$\mathrm{AICC}(p) = -2\log L(\hat{\phi}|y) + 2(p+1)n/(n-p-2),$$

BIC/Schwartz:

$$\mathrm{BIC}(p) = -2\log L(\hat{\phi}|y) + 2(p+1)\log n,$$

MDL: designed to maximize the compression of the data

$$\mathrm{M}DL(p) = -2\log L(\hat{\phi}|y) + (p+1)\log n,$$

Model Selection

Problem 1: How does one choose the *best* model?

Problem 2: What do we mean by *best*?

Some Objectives of Model Selection.

1. Choose the correct order model (*consistency*).
   - There exists a *true* model and the model selection procedure will choose the *correct* set of covariates and the right family of covariance functions as sample size increases. (BIC,MDL)
2. Choose the model that performs *best* for prediction (*efficiency*).
   - Find the model that predicts (or interpolates) well at unobserved locations. (AIC, AICC)
3. Choose the model that *maximizes* data compression.
   - Find a model that summarizes the data in the most compact fashion, yet retains the salient features present in the data. (MDL)

Fitting Piecewise AR Models

Time Series: $y_1, \ldots, y_n$

Piecewise AR model:

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

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

τ_j = location of j^{th} break point

γ_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

σ_j = scale in j^{th} epoch

Model Selection Using Minimum Description Length

Basics of MDL (Rissanen):

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

Illustration Using a Simple Regression Model (see T. Lee `01)

Encoding the data: $(x_1,y_1), \ldots, (x_n,y_n)$

1. "Naïve" case

$$L("naive") = L(x_1,\ldots,x_n) + L(y_1,\ldots,y_n)$$
$$= L(x_1)+\cdots+L(x_n) + L(y_1)+\cdots+L(y_n)$$

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 $\{\varepsilon_i\} \sim$ IID $N(0,\sigma^2)$. Then

$$L("p=1") = L(x_1)+\cdots+L(x_n) + \underbrace{L(\hat{a}_0)+L(\hat{a}_1)+L(\hat{\sigma}^2)+L(\hat{\varepsilon}_1,\ldots,\hat{\varepsilon}_n \mid \hat{a}_0,\hat{a}_1,\hat{\sigma}^2)}_{A}$$

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:

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

After some calculation, the MDL for this class of models is

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

$$= \log_2 m + m \log_2 n + \sum_{j=1}^{m} \log_2 p_j + \sum_{j=1}^{m} \frac{p_j + 2}{2} \log_2 n_j + \sum_{j=1}^{m} \frac{n_j}{2} \log_2 (2\pi \hat{\sigma}_j^2) + \frac{n}{2}$$

Goal: minimize MDL wrt to

m = number of segments

τ_j = location of j^{th} break point

γ_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

σ_j = scale in j^{th} epoch

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

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

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, -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. .

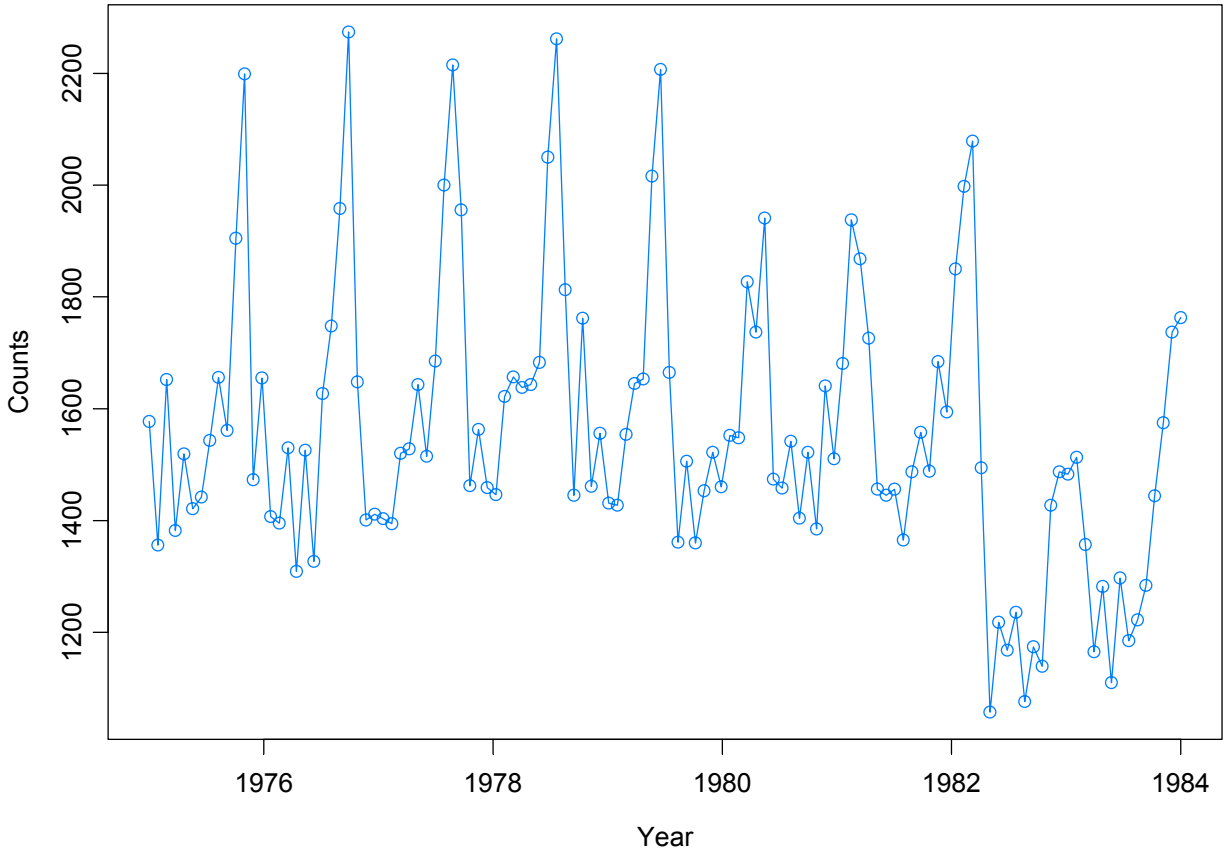
Various Strategies:

- 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*.

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).



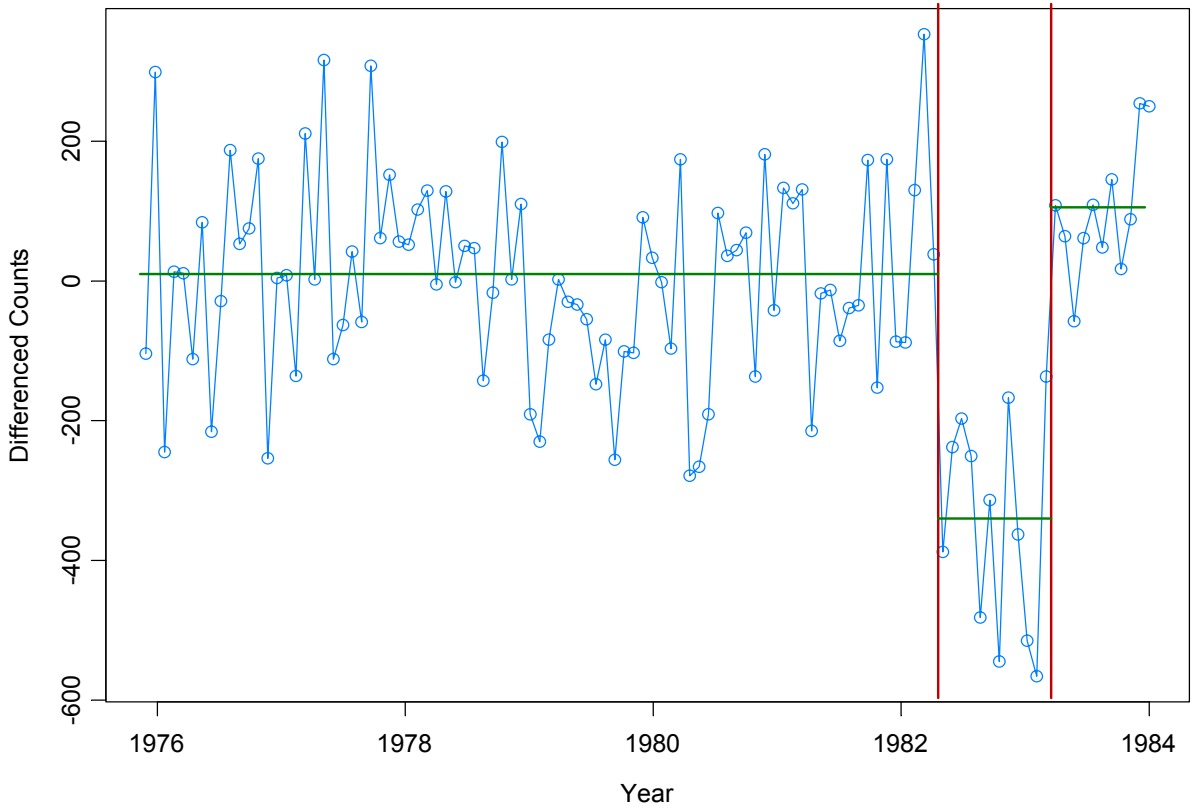
$$Y_t = a + bf(t) + W_t,$$

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

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); Plot below is for differenced series, $Y_t - Y_{t-12}$

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)

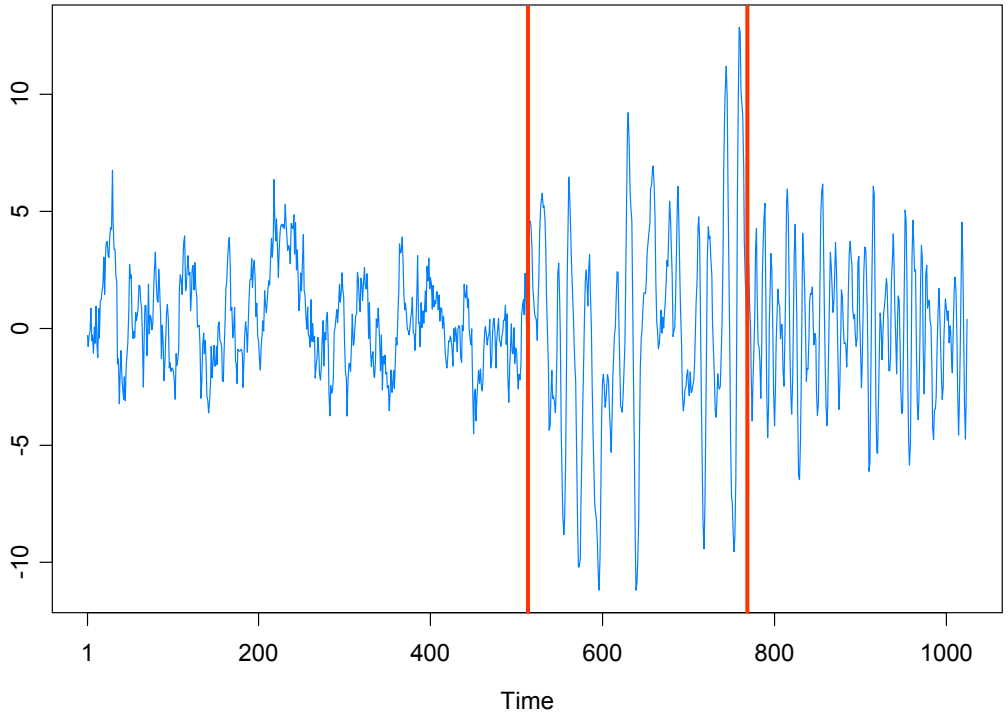
Simulation Examples-based on Ombao et al. (2001) test cases

1. Piecewise stationary with dyadic structure:

Consider a time series following the model,

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

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

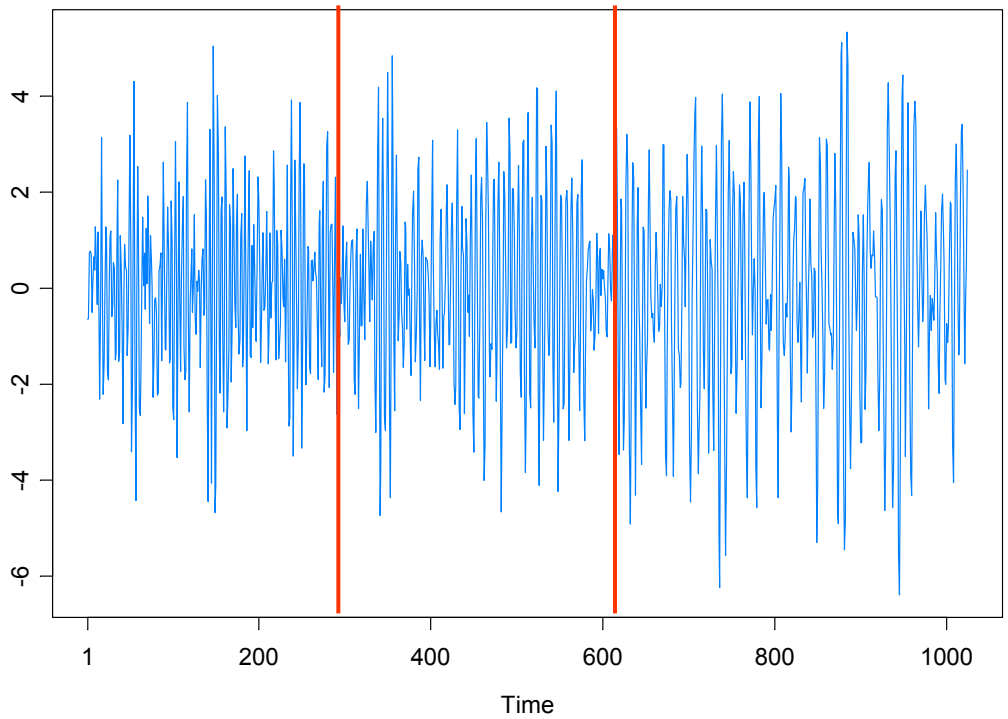


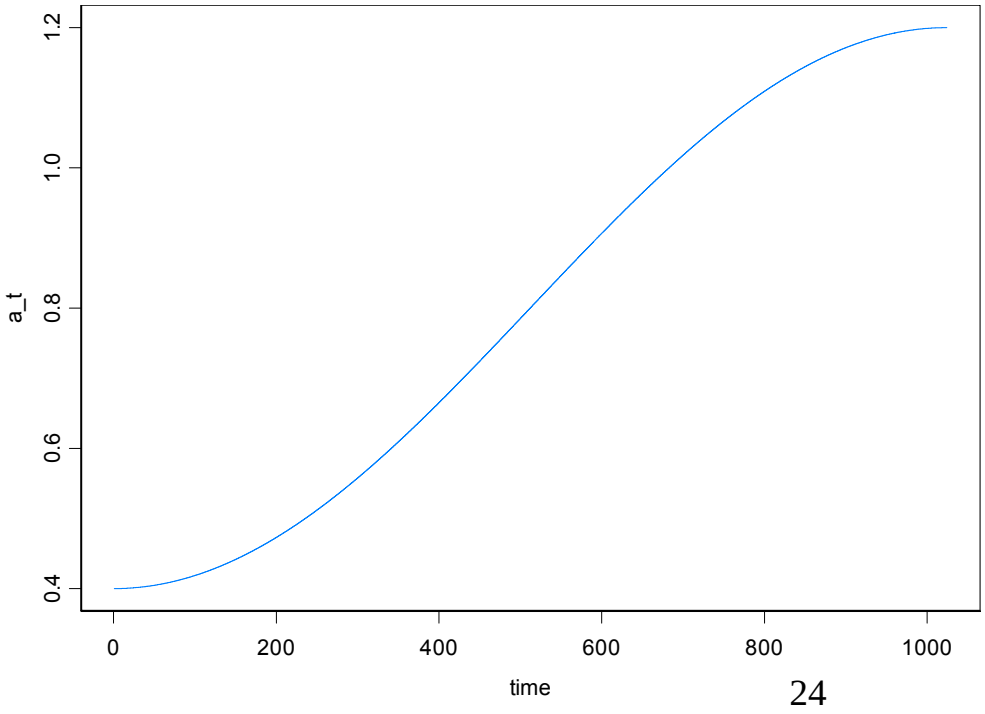
Simulation Examples (cont)

2. Slowly varying AR(2) model:

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

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





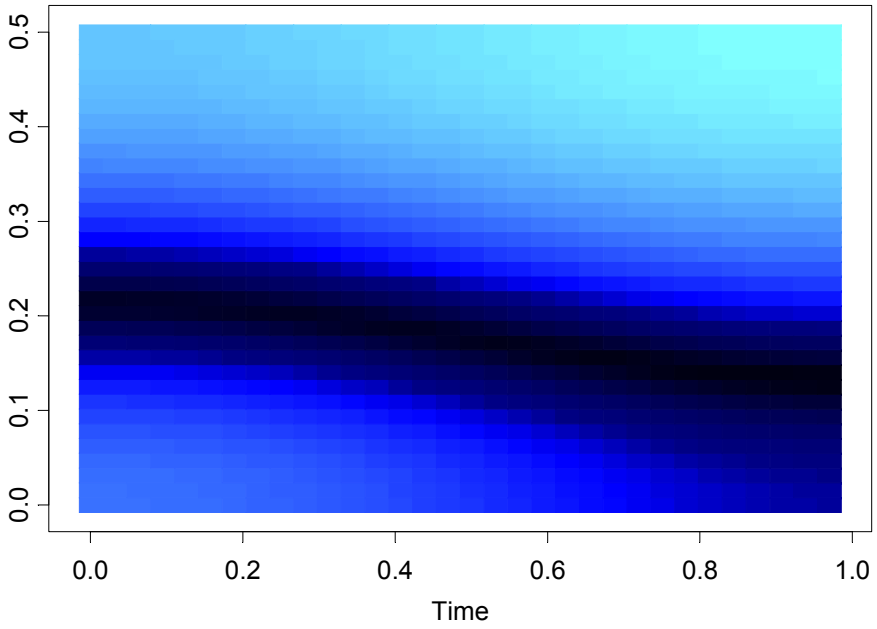
2. Slowly varying AR(2) (cont)

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

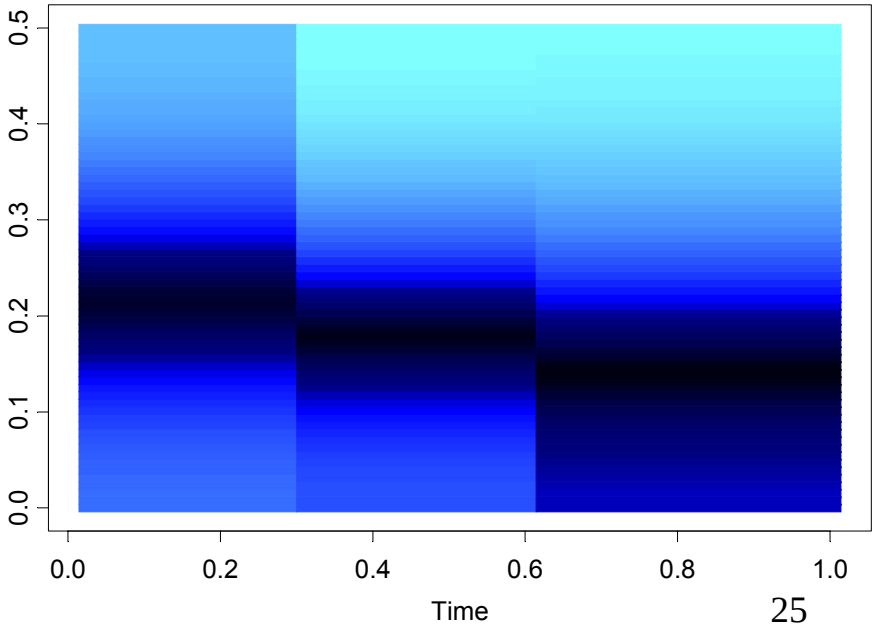
Fitted model:

	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



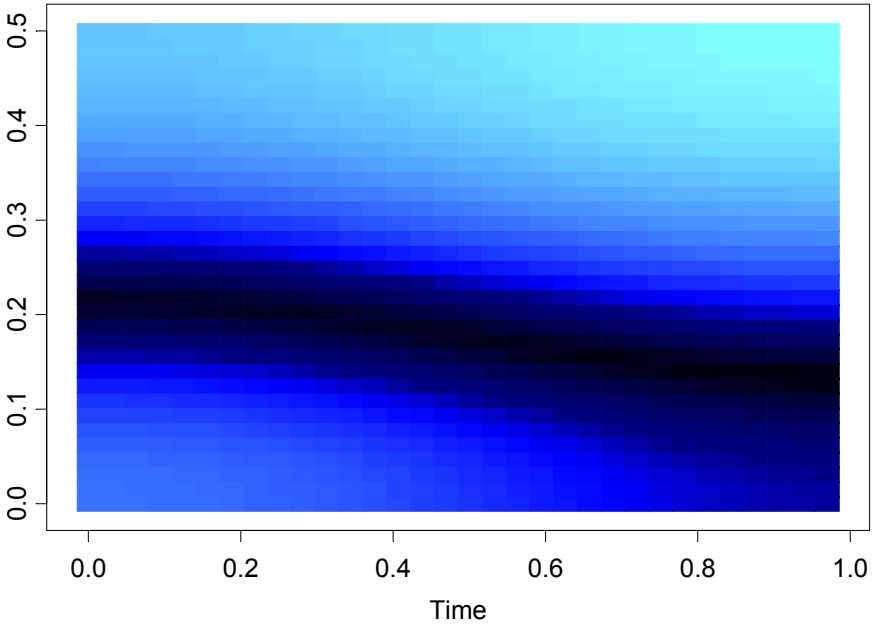
Fitted Model



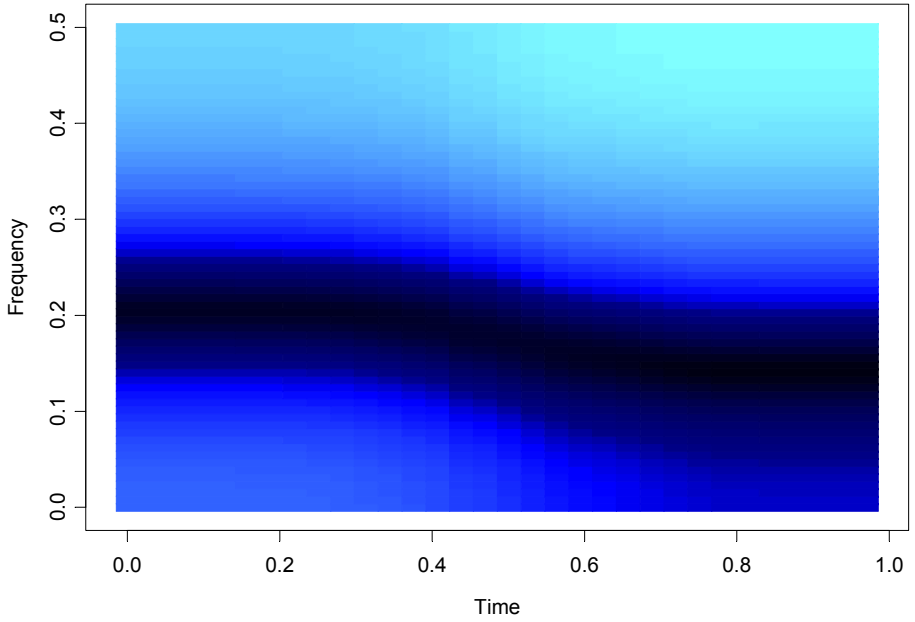
2. Slowly varying AR(2) (cont)

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

True Model

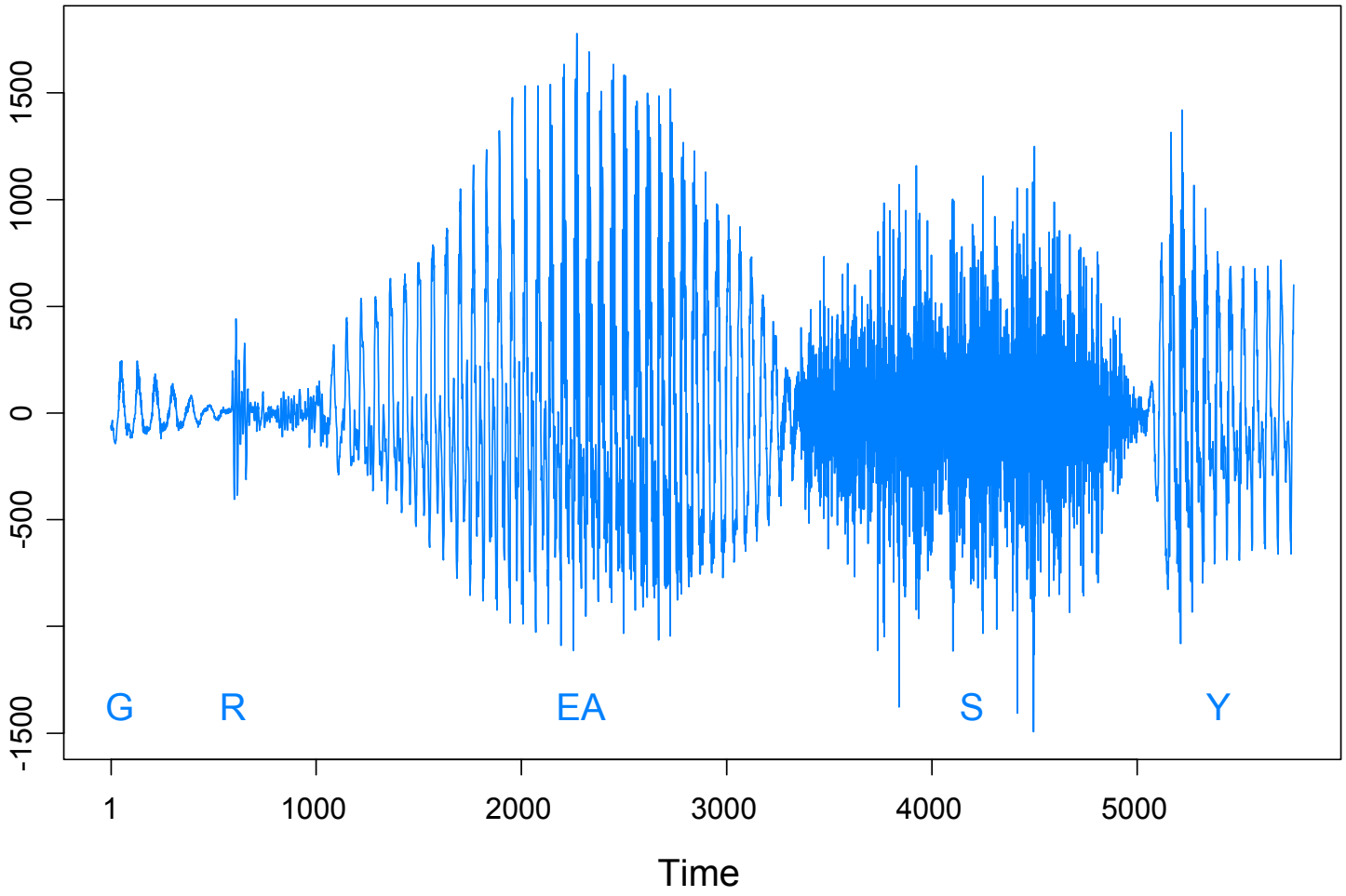


Average Model



Examples

Speech signal: GREASY

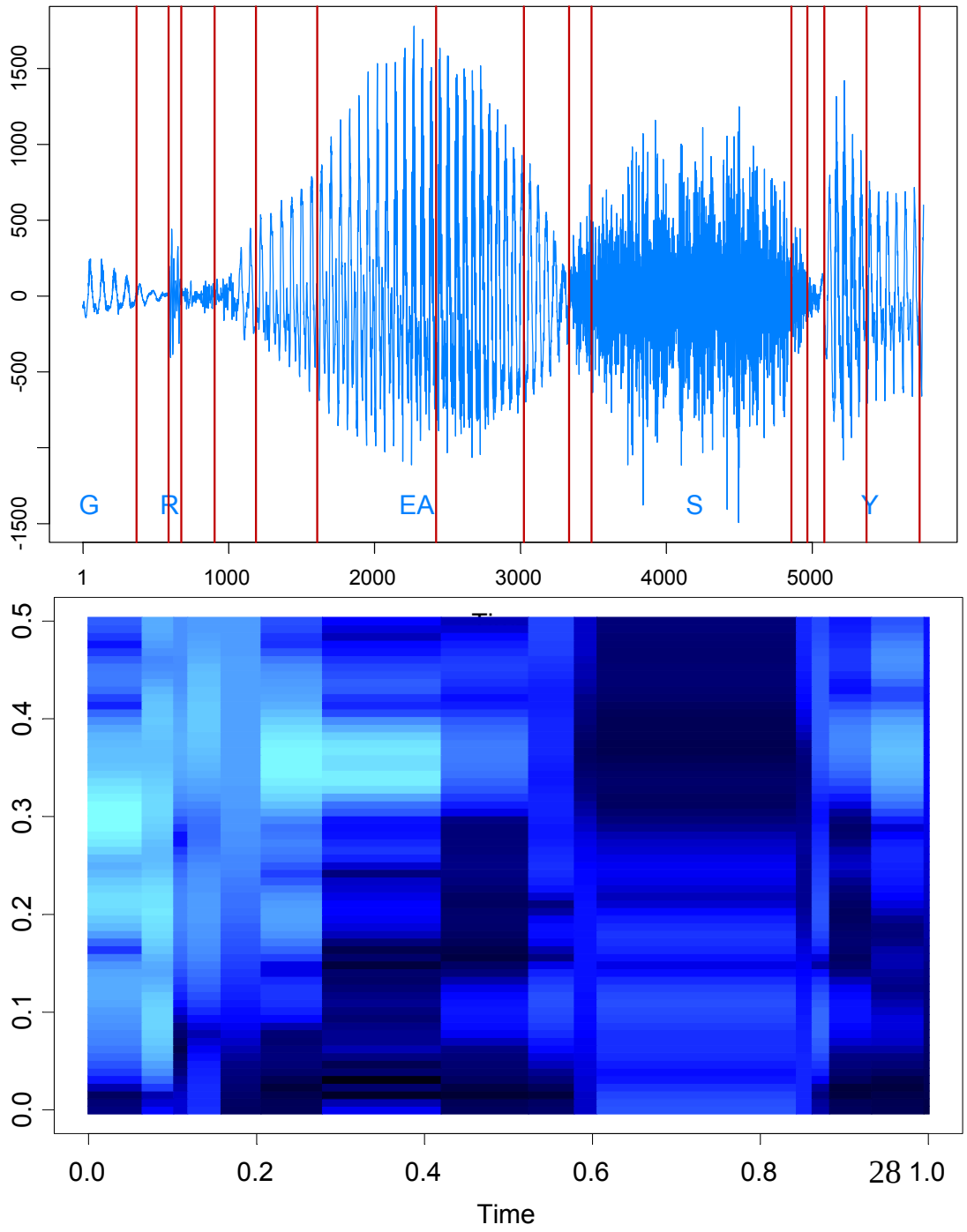


Speech signal: GREASY

n = 5762 observations

m = 15 break points

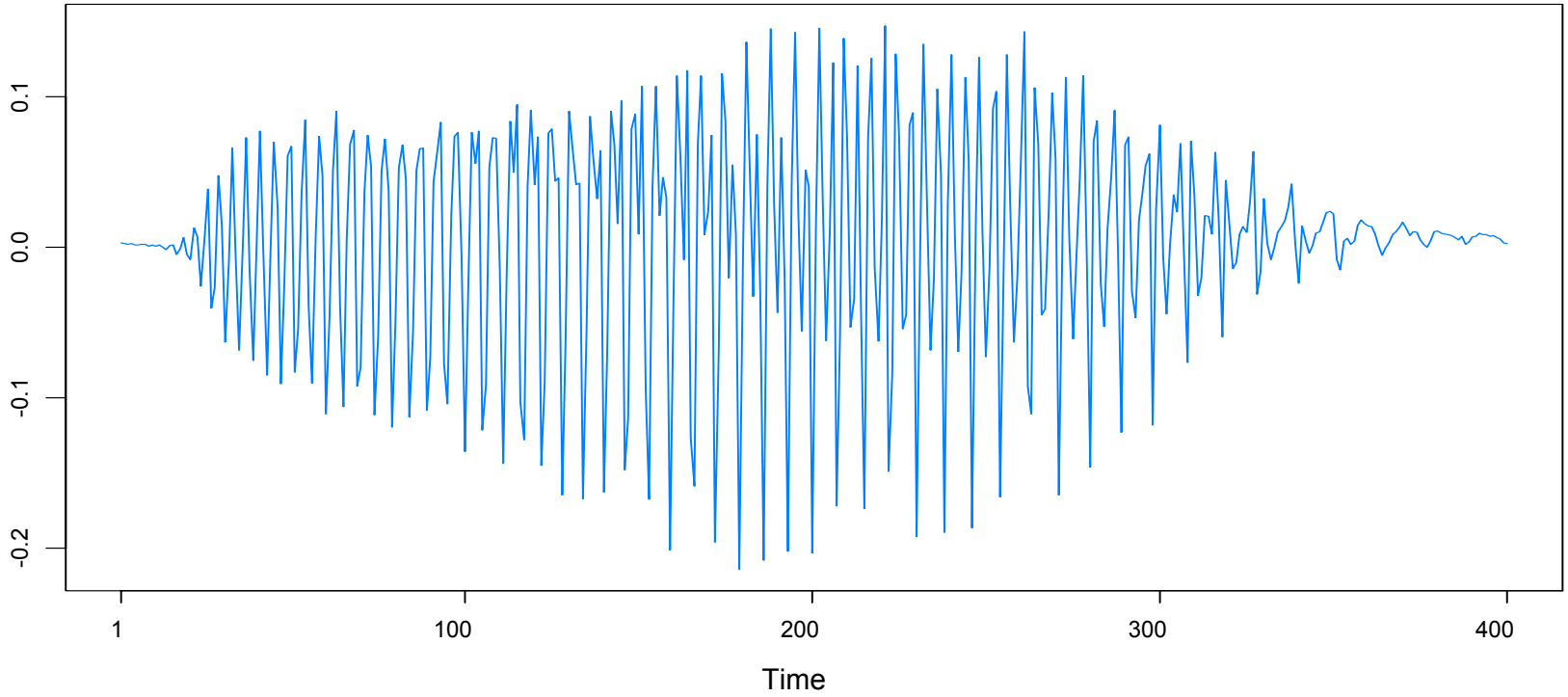
Run time = 18.02 secs



Examples

Large brown bat echolocation: 400 data points taken at 7microsecond 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



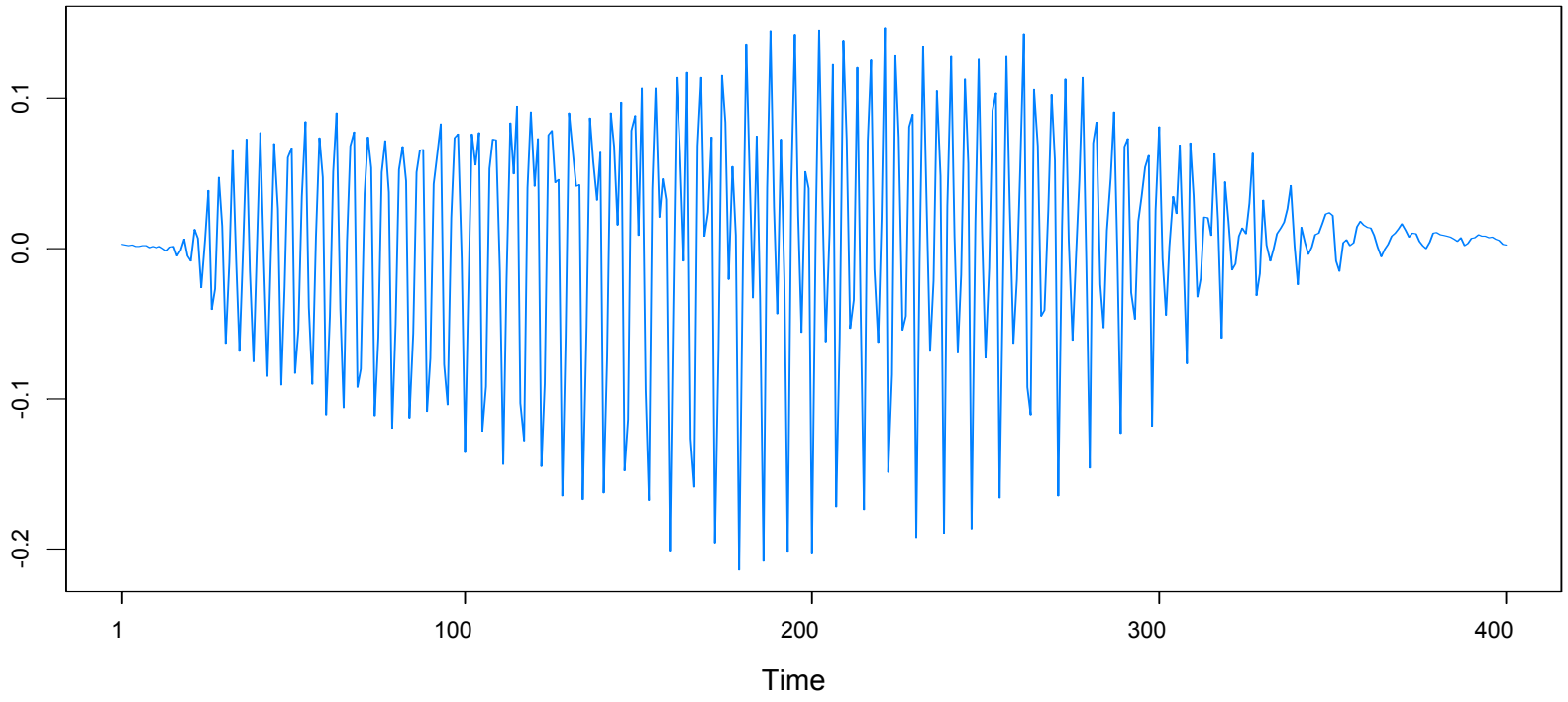
Features of data:

• *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.

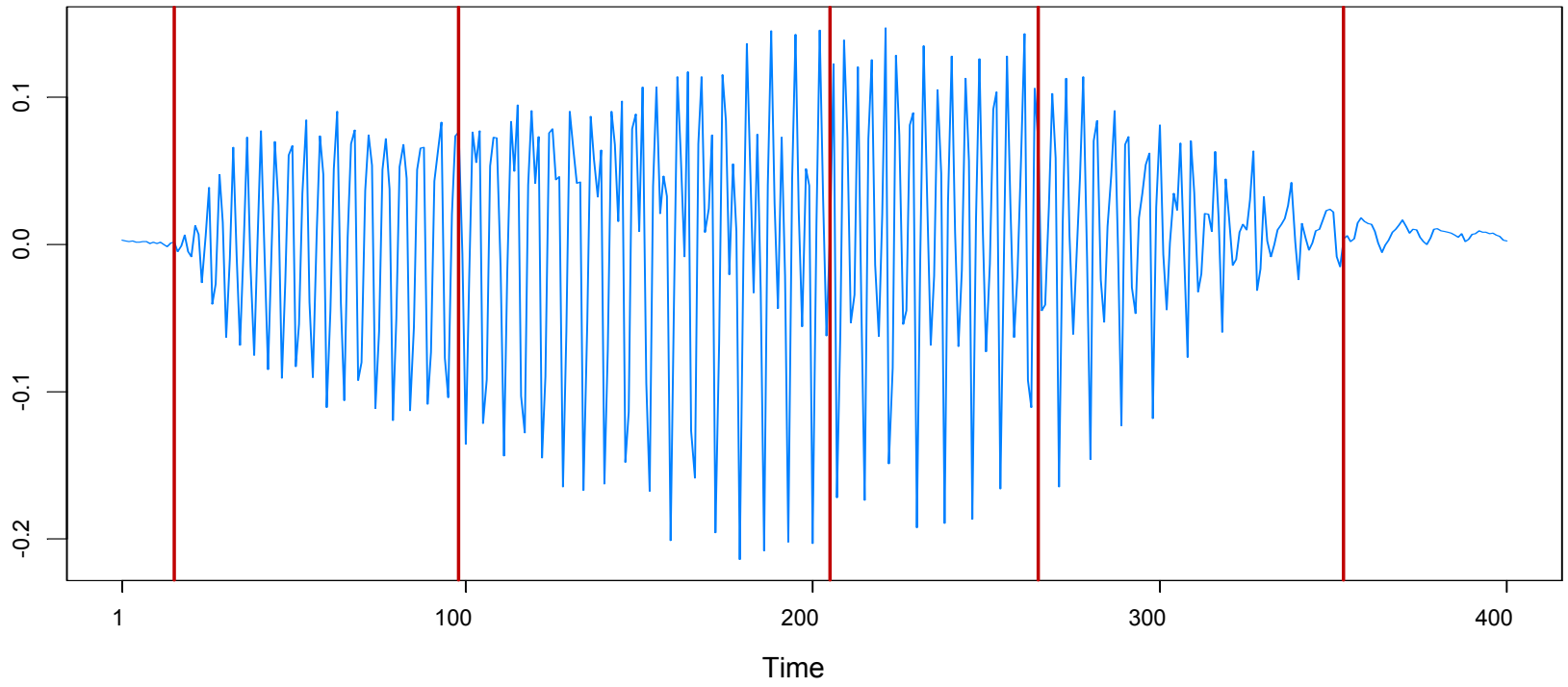
• an AR(20) model is the *best fitting* AR model. Residuals are *uncorrelated* but *not independent*.



Examples (bat data cont)

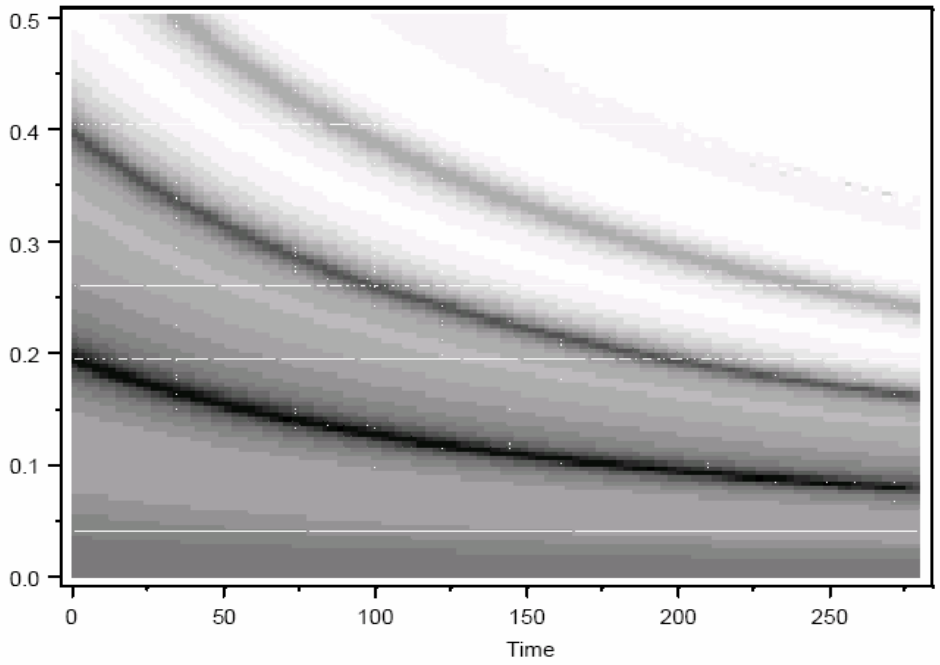
GA results: 6 pieces, breaks at τ_1=16, τ_2=98, τ_3= 205, τ_4= 265, τ_5= 353.

Fitted model: AR orders 1, 6, 13, 7, 13, 5; Total run time 4.7 secs

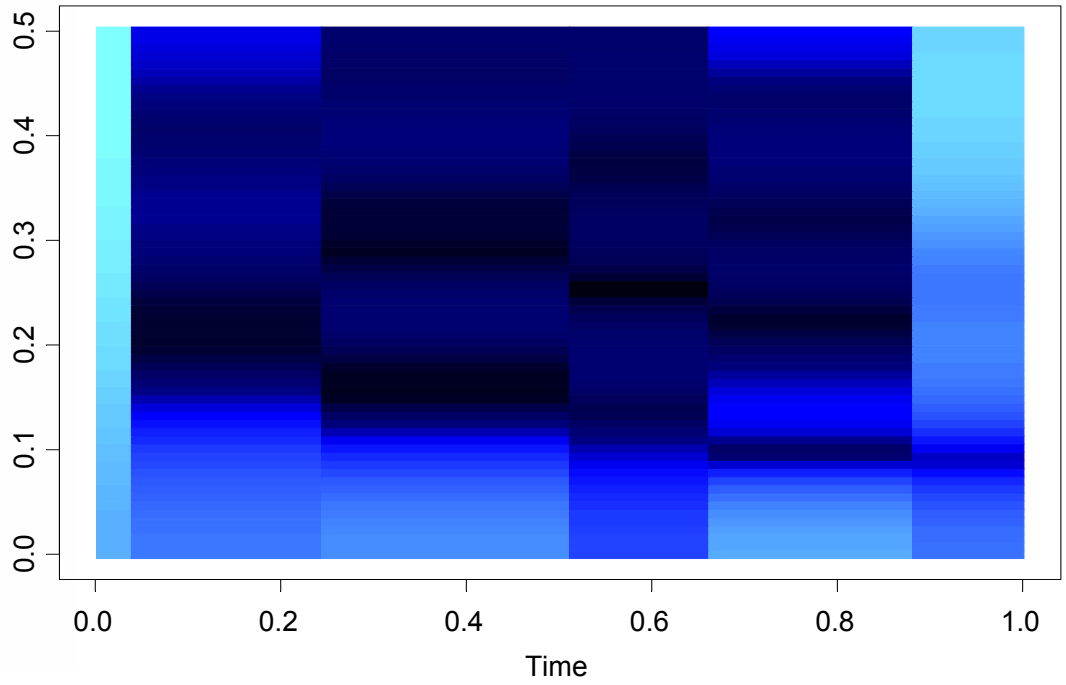


Examples (bat data spectrograms)

Euler(12), Gray et al



Auto-PARM

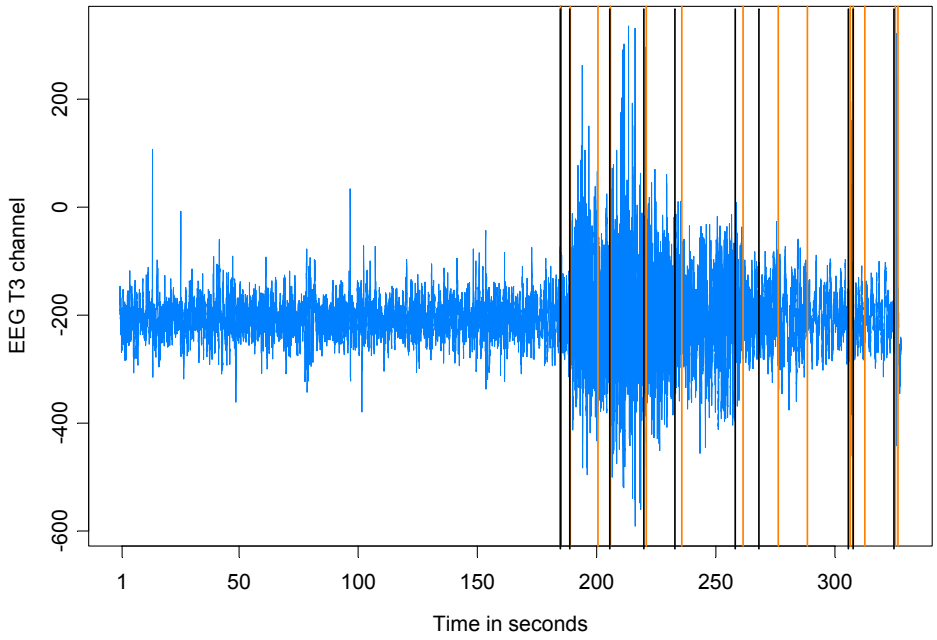


Example: EEG Time series

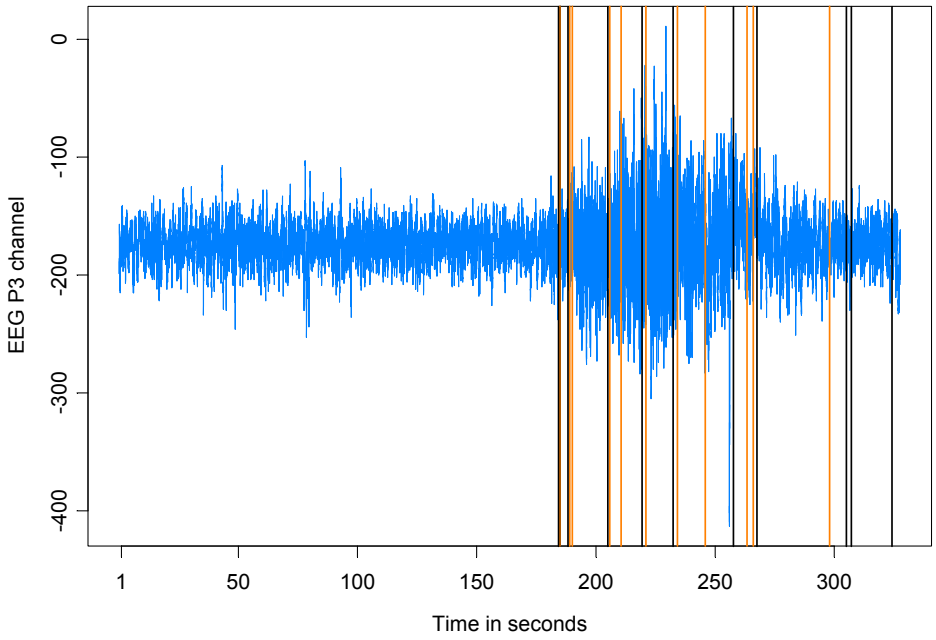
Data: Bivariate EEG time series at channels T3 (left temporal) and P3 (left parietal). Female subject was diagnosed with left temporal lobe epilepsy. Data collected by Dr. Beth Malow and analyzed in Ombao et al (2001). (n=32,768; sampling rate of 100H). Seizure started at about 1.85 seconds.

GA bivariate results: 14 breakpoints for T3; 11, 7, 2, 6, 15, 2, 3, 5, 9, 5, 4, 1 breakpoints for P3

T3 Channel



P3 Channel

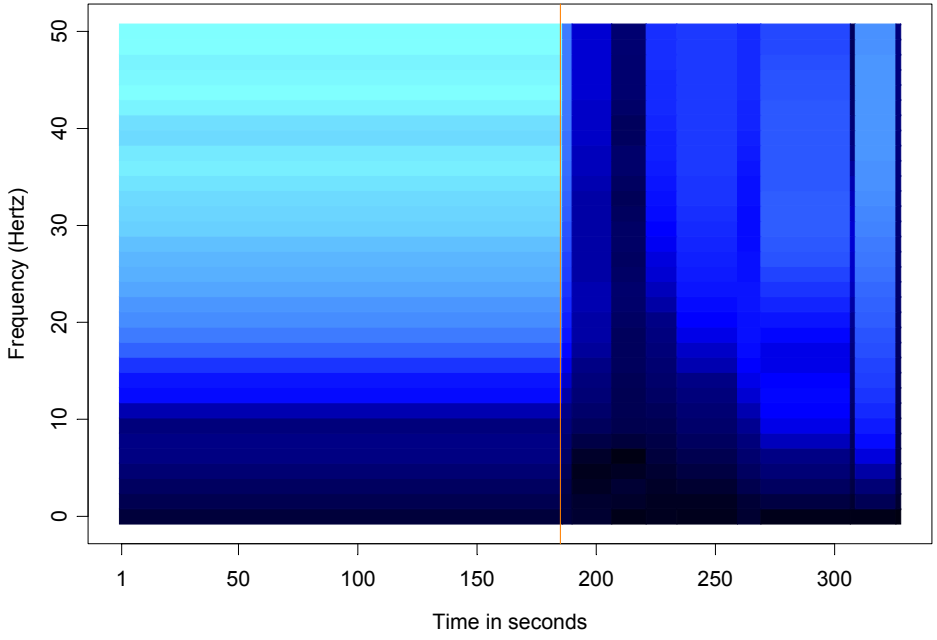


Example: EEG Time series (cont)

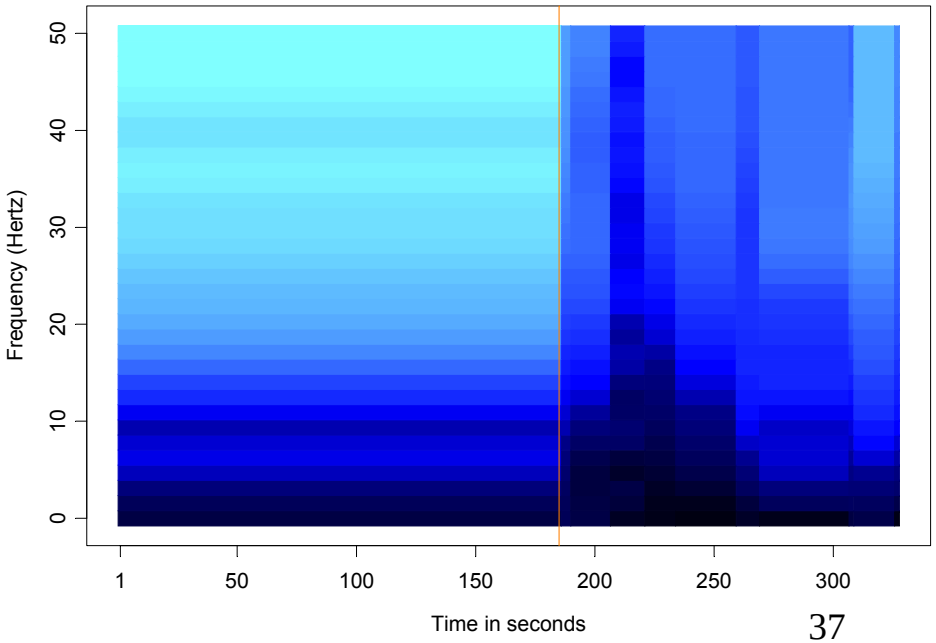
Remarks:

- 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.

T3 Channel



P3 Channel

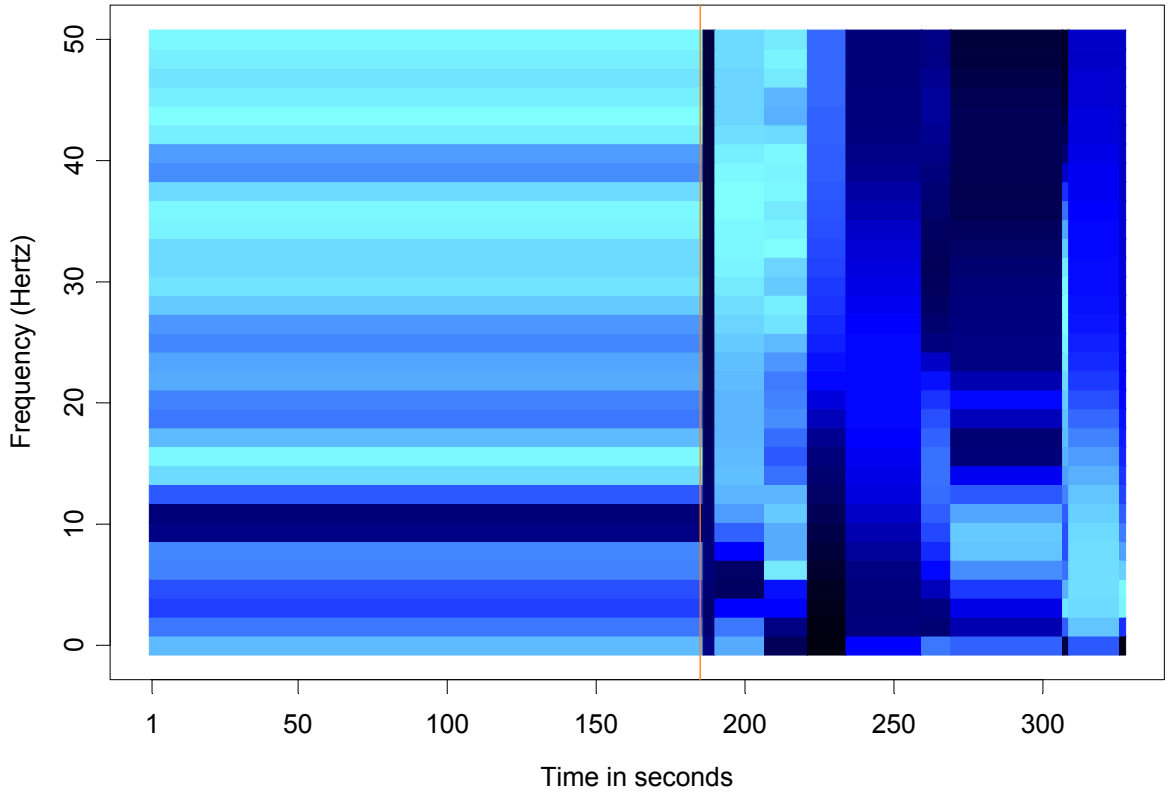


Example: EEG Time series (cont)

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.

T3/P3 Coherency



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

Parameters:

m = number of break points

τ_k = location of break points

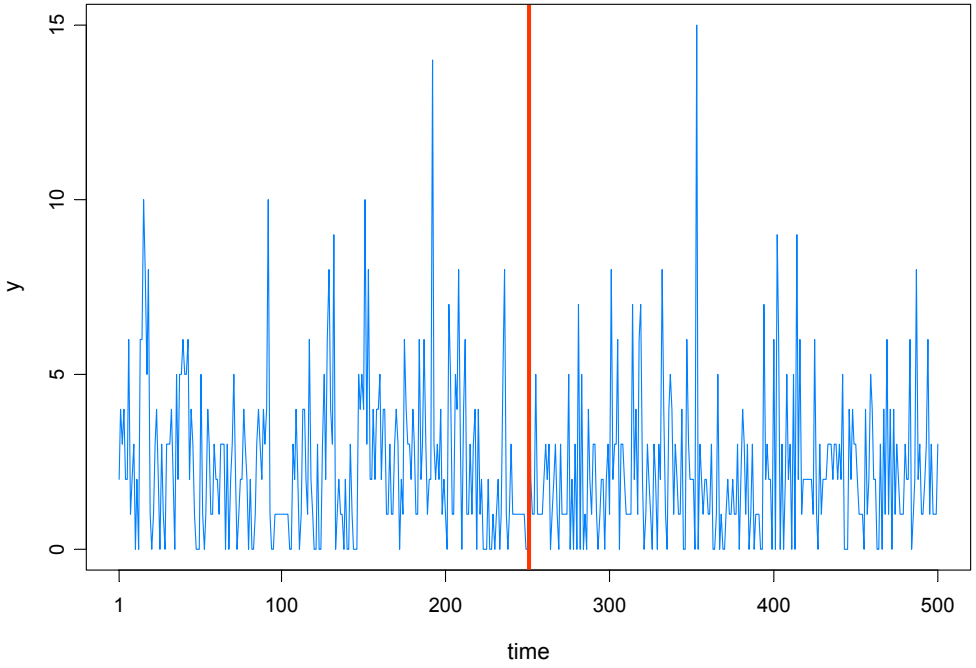
γ_k = level in k^{th} epoch

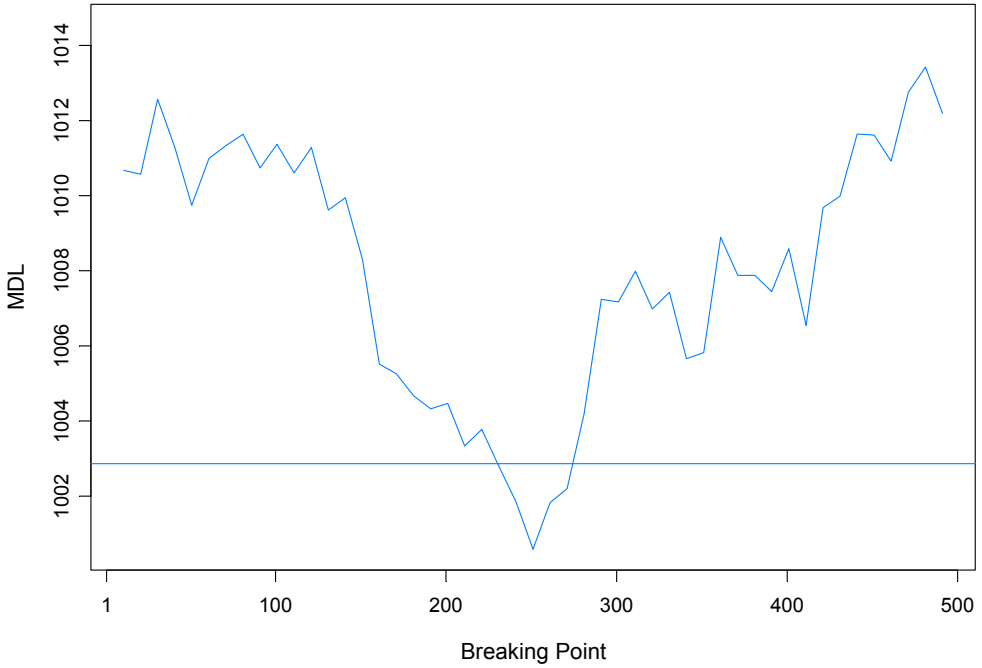
ϕ_k = AR coefficients k^{th} epoch

σ_k = scale in k^{th} epoch

Count Data Example

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



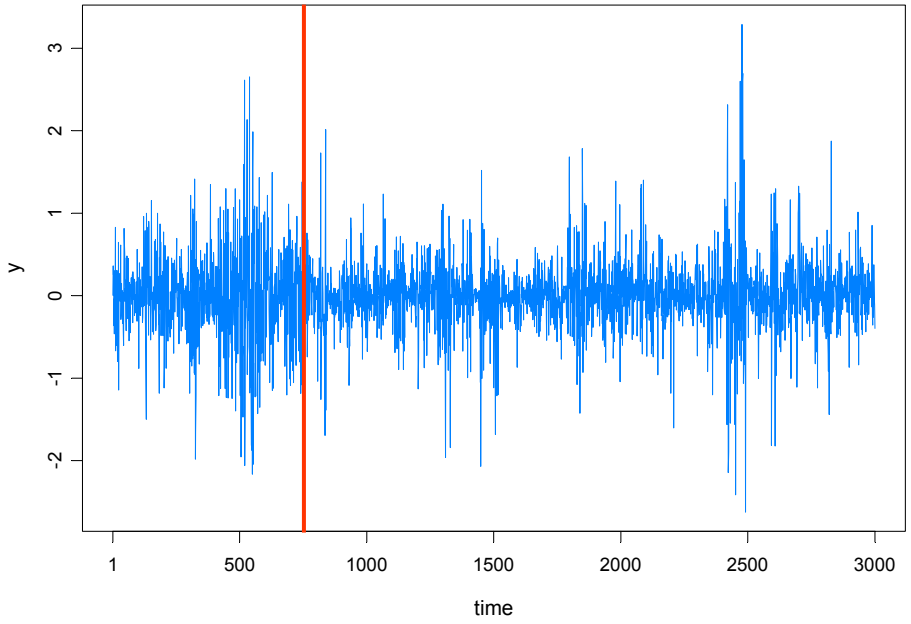


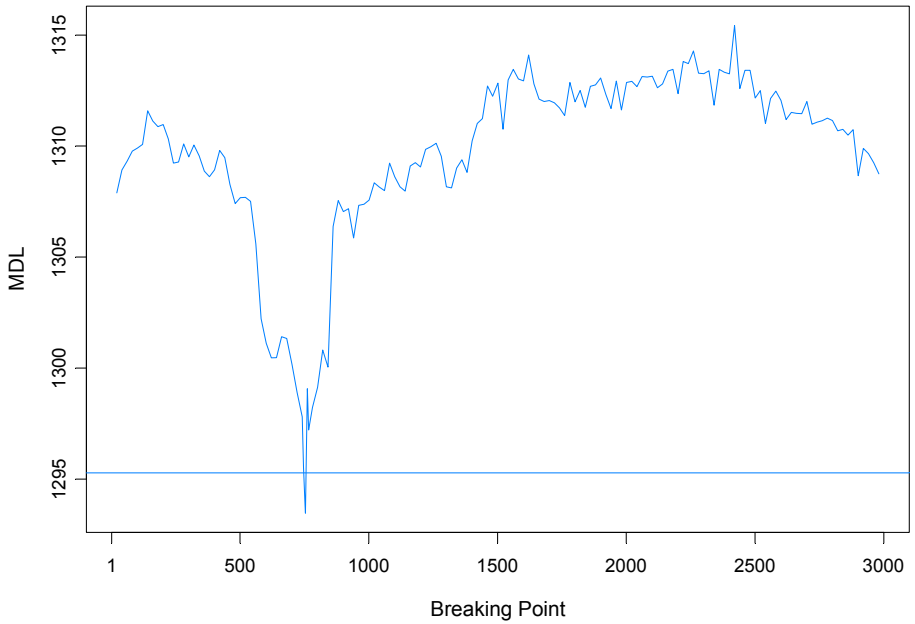
True model:

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

SV Process Example

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



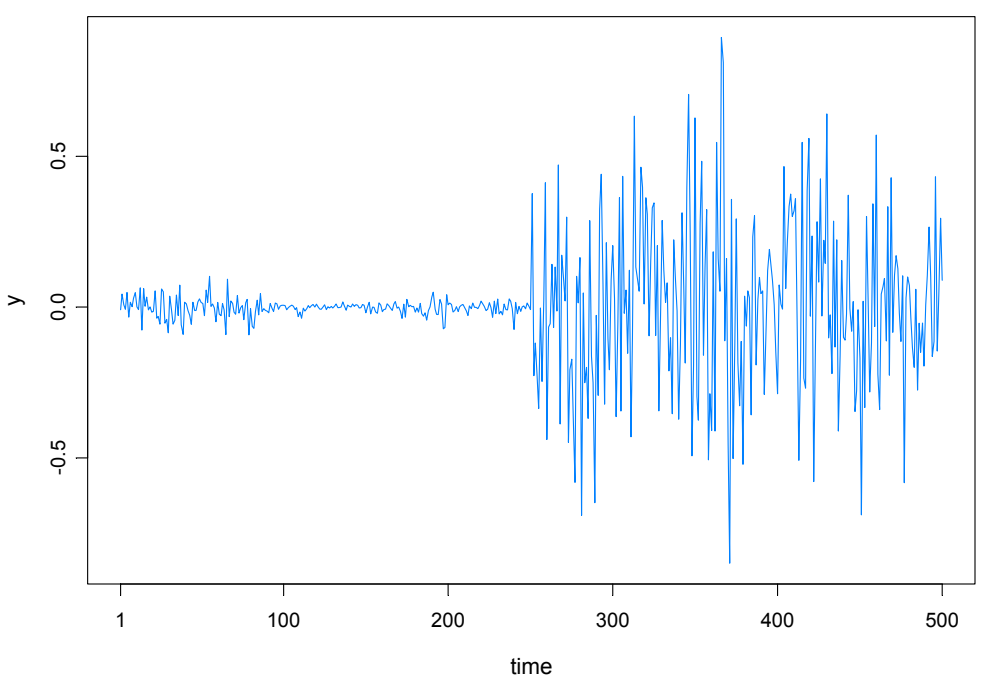


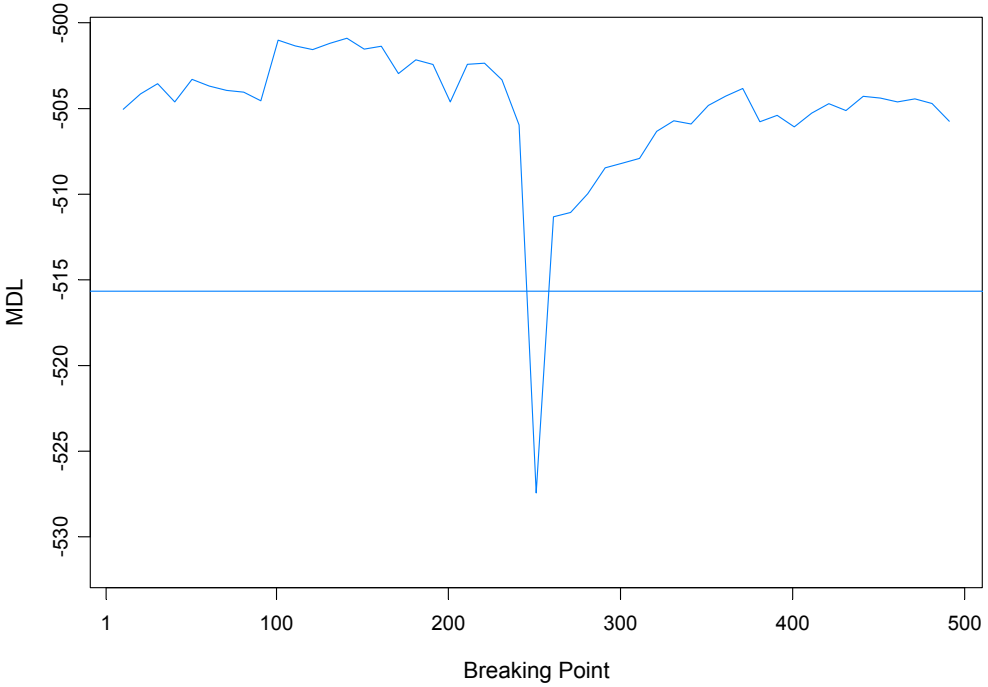
True model:

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

SV Process Example

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





True model:

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

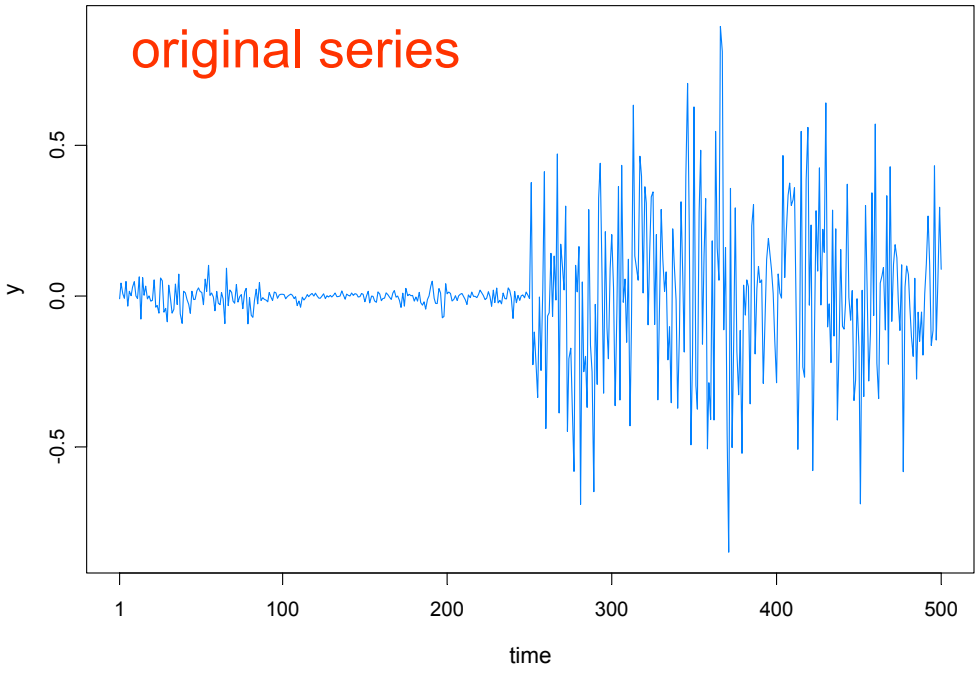
SV Process Example-(cont)

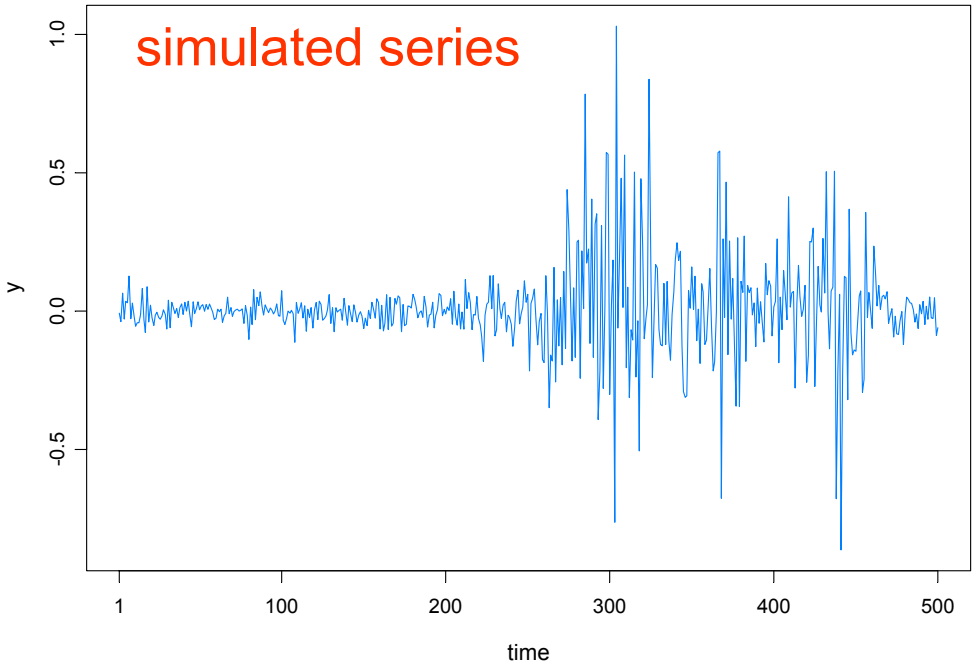
True model:

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

Fitted model based on no structural break:

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

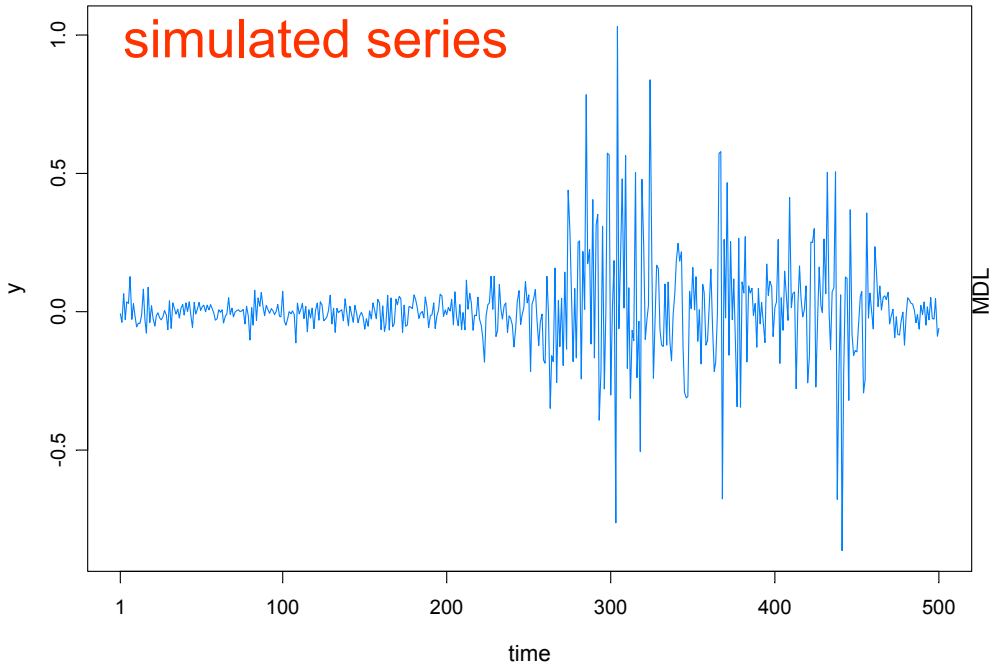


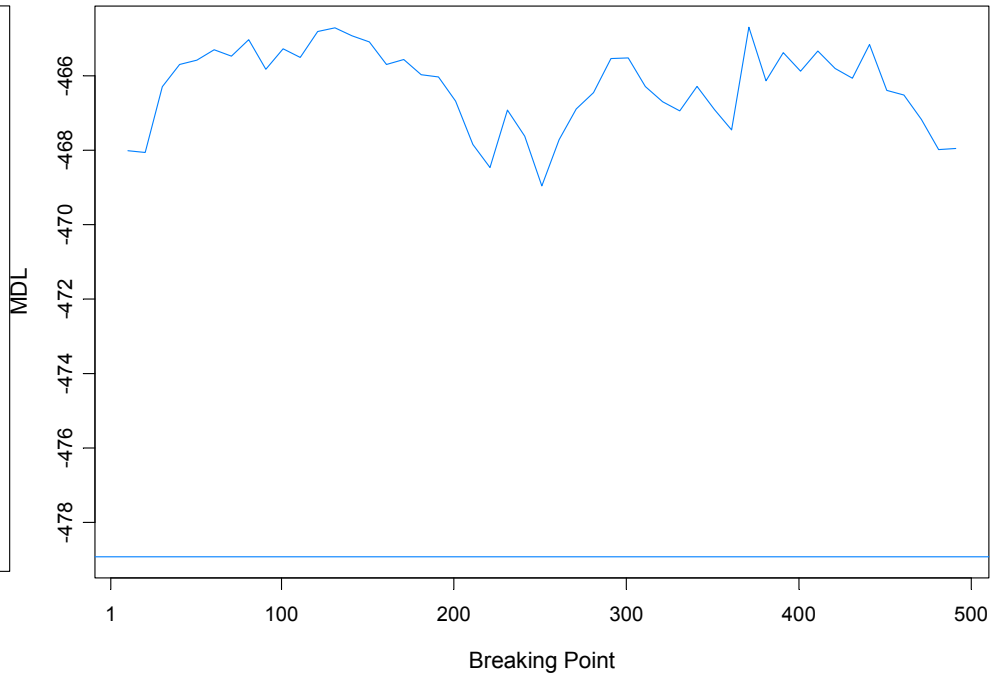


SV Process Example-(cont)

Fitted model based on no structural break:

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





Summary Remarks

1. *MDL* has an appealing model selection interpretation that may be useful in a variety of applications.

2. *MDL* appears to be a good criterion for detecting structural breaks.

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

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