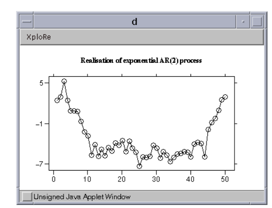
Web quantlets for time series analysis

Wolfgang Härdle, Torsten Kleinow, Rolf Tschernig

(this paper online as pdf)

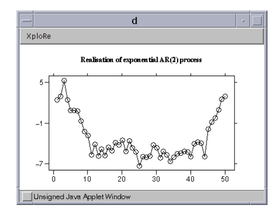


1. Introduction

"Empower people through great software anytime, anywhere and on any device"

Bill Gates

"Empower statisticians through great methods anytime, anywhere and on any device"

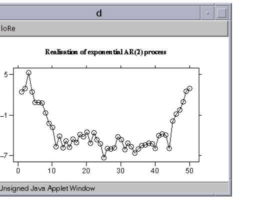


New Methods in Time Series Analysis involve often

- complex computing algorithms
- hard mathematical problems

Example: nonparametric time series analysis

$$Y_t = f(Y_{t-i_1}, Y_{t-i_2}, \ldots, Y_{t-i_m}) + \sigma(Y_{t-i_1}, Y_{t-i_2}, \ldots, Y_{t-i_m})\xi_t$$



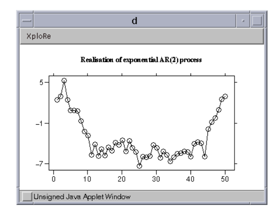
Example (ctd.)

$$Y_t = f(Y_{t-i_1}, Y_{t-i_2}, \ldots, Y_{t-i_m}) + \sigma(Y_{t-i_1}, Y_{t-i_2}, \ldots, Y_{t-i_m})\xi_t$$

- computational difficulty: lag selection
- mathematical difficulty: nonparametric smoothing

Method of CAFPE solves this problem but is computationally and mathematically difficult. It is, however, very interesting for detection of new models, see e.g.

Tschernig, R. & Yang, L. (2000), 'Nonparametric lag selection for time series', Journal of Time Series Analysis.



Solutions:

do it yourself 

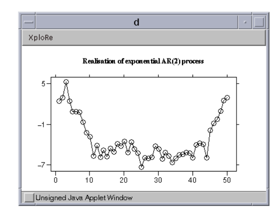
telephone 

email help.me@colleague

method cemetery 

Java

Web Quantlets



Plan of talk

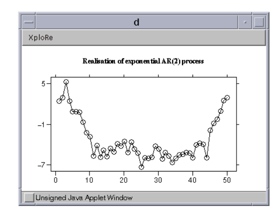
√ 1. Introduction

2. Nonparametric identification of nonlinear time series models

3. Web quantlets

4. Illustration

5. ebook



2. Nonparametric identification of nonlinear time series models

Nonlinear conditional heteroskedastic autoregressive time series process

$$Y_t = f(\mathbf{X}_t) + \sigma(\mathbf{X}_t)\xi_t, \quad t = 0, 1, \dots,$$

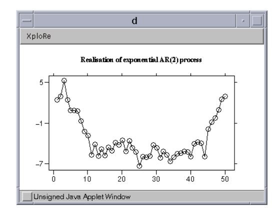
with

$$\mathbf{X}_t = (Y_{t-i_1}, Y_{t-i_2}, \dots, Y_{t-i_m})^T$$

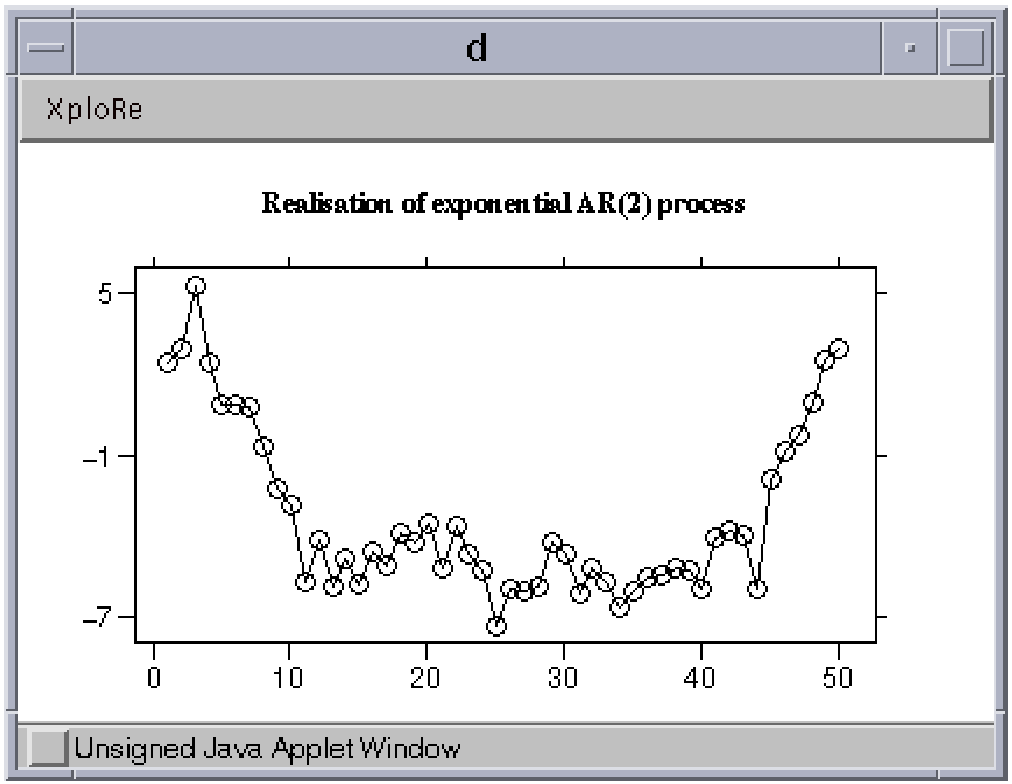
and

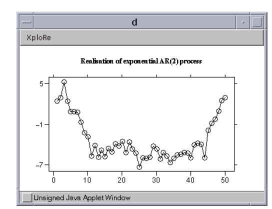
$$\xi_t \text{ i.i.d}, \quad E[\xi_t] = 0, \quad E\left[\xi_t^2\right] = 1$$

used in establishing new time series models



Nonparametric identification of nonlinear time series models



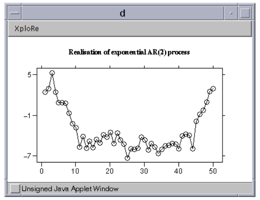


Nonparametric identification of nonlinear time series models

requires two steps:

1) **lag selection**: choose the relevant lags $i_1, \ldots, i_m$ including their number m for the autoregression function f and the conditional standard deviation σ.

2) **function estimation**: estimate f and, if desired, σ for the chosen lag vector.



Step 2: Function estimation

For given set of lags: $i_1, i_2, \ldots, i_m$:

Local linear function estimation $\widehat{f}(\mathbf{x}, h) = \widehat{c}_0$

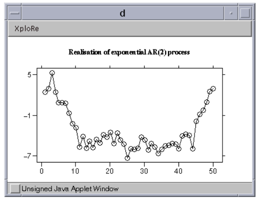
$$\{\widehat{c}_0, \widehat{\mathbf{c}}\} = \arg\min_{\{c_0, \mathbf{c}\}}$$

$$\sum_{t=1}^{T} \left\{ Y_t - c_0 - (\mathbf{X}_t - \mathbf{x})^T \mathbf{c} \right\}^2 K_h(\mathbf{X}_t - \mathbf{x})$$

with product kernel

$$K_h(\mathbf{X}_t - \mathbf{x}) = h^{-m} \prod_{i=1}^{m} K\left\{(X_{t,i} - x_i)/h\right\}$$

(Härdle & Tsybakov, 1997)



Step 2: Function estimation - bandwidth choice

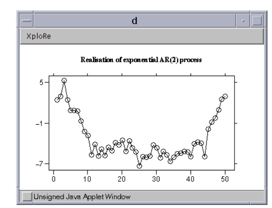
Final Prediction Error (FPE)

$$FPE(h, i_1, \ldots, i_m) = E\left[\left(\breve{Y}_t - \widehat{f}(\breve{X}_t, h)\right)^2 w(\breve{X}_{t,M})\right]$$

where $\breve{X}_{t,M} = (\breve{Y}_{t-1}, \ldots, \breve{Y}_{t-M})^T$.

Under appropriate conditions it holds that

$$FPE(h) = AFPE(h) + o\{h^4 + T^{-1}h^{-m}\}$$



Asymptotic Final Prediction Error

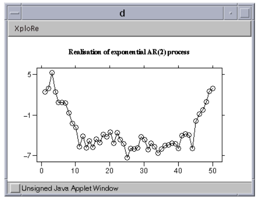
Under appropriate conditions it holds that

$$FPE(h) = AFPE(h) + o\{h^4 + T^{-1}h^{-m}\}$$

where

$$AFPE(h) \quad = \quad A \quad + \quad b(h)B \quad + \quad c(h)C$$

$$\text{Asymp. FPE} \quad = \quad \int \text{Var} \quad + \quad \int Var(\widehat{f}) \quad + \quad \int bias(\widehat{f})^2$$



Step 2: Function estimation - bandwidth choice

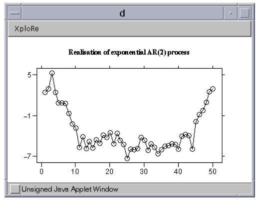
Plug-in bandwidth

Estimate asymptotically optimal bandwidth

$$h_{opt} = \left(d(T, m, K(\cdot)) \frac{B}{C} \right)^{\frac{1}{m+4}}$$

by estimating B and C

e.g. by using direct local quadratic estimators for C (Yang & Tschernig, 1999)



Step 1: Lag selection (Tschernig & Yang, 2000)

Estimated Asymptotic FPE

$$AFPE = \widehat{A}(\widehat{h}_{opt}) + 2K(0)^m T^{-1} \widehat{h}_{opt}{}^{-m} \widehat{B}(\widehat{h}_B)$$

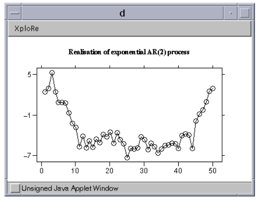
Estimated Corrected Asymptotic FPE (CAFPE)

$$CAFPE = AFPE \left\{ 1 + mT^{-4/(m+4)} \right\}$$

Select lag vector with smallest $AFPE$ or $CAFPE$.

Both criteria are consistent, e.g.

$$Pr\left(\widehat{m} = m, \widehat{i_s} = i_s, s = 1, 2, \ldots, m\right) \stackrel{T\to\infty}{\to} 1.$$



Step 1: Lag selection - Details

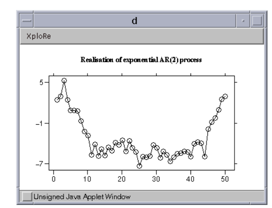
Details of *AFPE* estimator

$$\widehat{A}(h) = T^{-1} \sum_{t=1}^{T} \left\{ y_t - \widehat{f}(\mathbf{X}_t, h) \right\}^2 w(\mathbf{X}_{t,M})$$

$$\widehat{B}(\widehat{h}_B) = T^{-1} \sum_{t=1}^{T} \left\{ Y_t - \widehat{f}(\mathbf{X}_t, \widehat{h}_B) \right\}^2 \frac{w(\mathbf{X}_{t,M})}{\widehat{\mu}(\mathbf{X}_t, \widehat{h}_B)}$$

and rule-of-thumb bandwidth

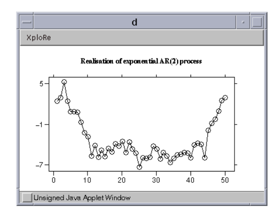
$$\widehat{h}_B = \widehat{\sigma} \left(\frac{4}{T+2} \right)^{1/(m+4)} T^{-1/(m+4)}$$



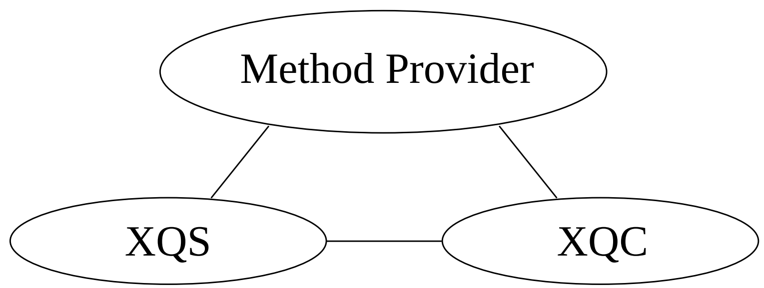
3. Web Quantlets

- Java applets
  - VESTAC project at KU Leuven
  - statlab Heidelberg
  - and many others
- Statistical programming language macros
  - StatLib
  - and many other professional sites

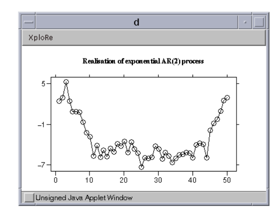
These are extremal positions.
We propose a "statistical middleware" solution based on a Client/Server concept.



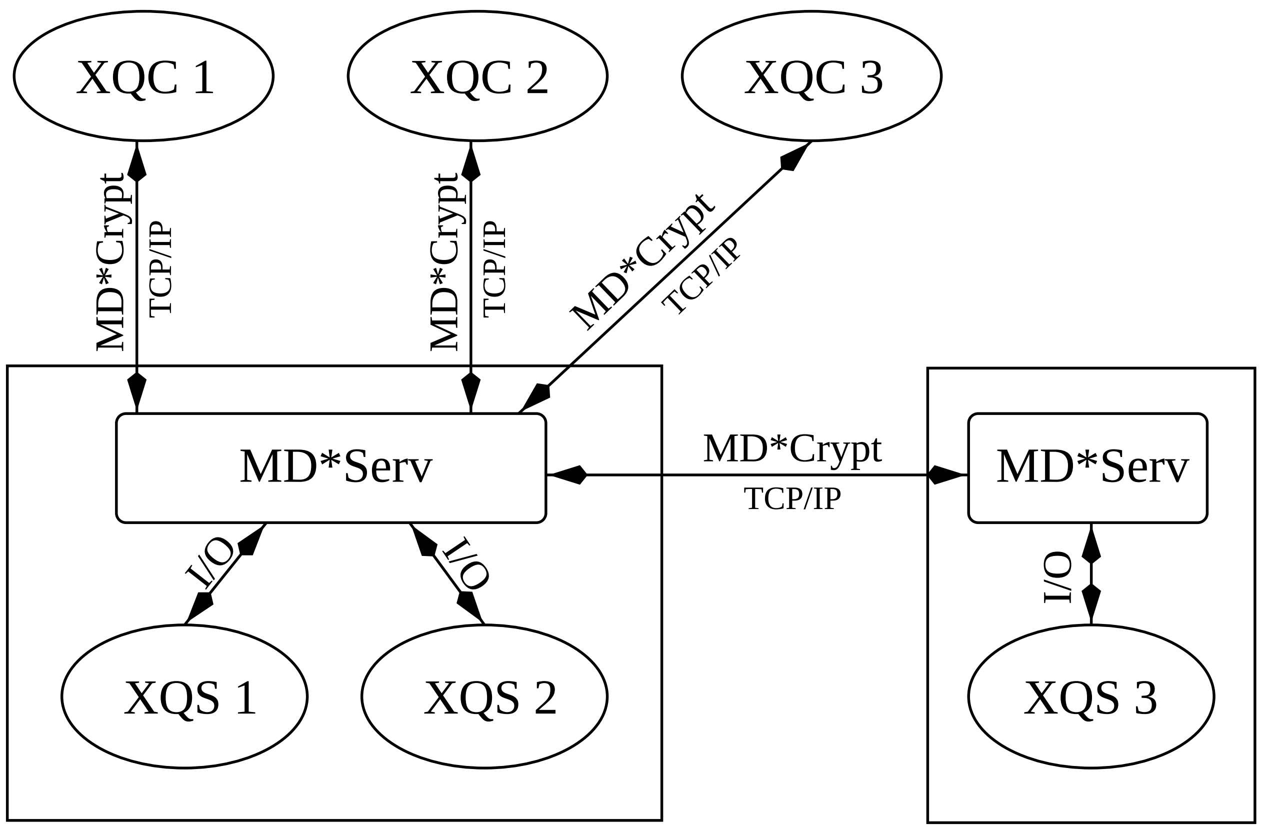
Basic Architecture

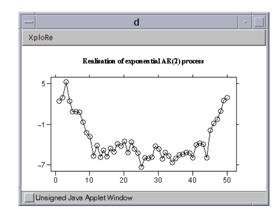


- client (XQC): student, scientist
- method provider: university, MD*Tech center
- server (XQS): university, private institution



Technical background





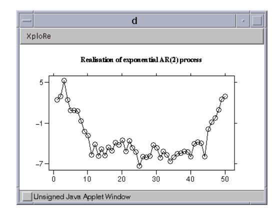
Elements of XQS/XQC

XQC client program, GUI, running on user's machine, depending on particular interests

MD*Crypt protocol for communication

MD*Serv middleware, manages the XQS/XQC communication

XQS server, provides statistical computing power



4. Illustration

- generate nonlinear time series data
- conduct nonparametric model identification
- present lag selection results
- do all via the XQS/XQC architecture

The example

- is contained in the cafpeexample which is written in the statistical language XploRe
- can be edited and executed from any standard Webbrowser

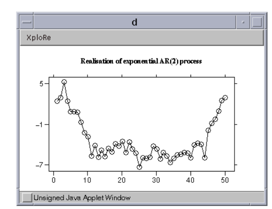


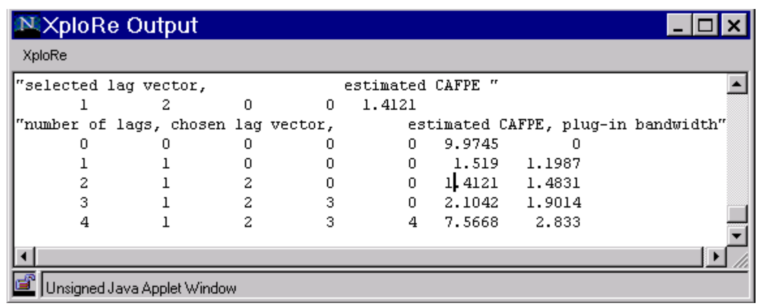
Illustration - lag selection

- generate 50 observations of exponential autoregressive process

  $$Y_t = 0.3Y_{t-1} + 0.6Y_{t-2} + (1.9Y_{t-1} - 1.1Y_{t-2})\exp(-0.1 * Y_{t-1}^2) + \xi_t$$

  using the quantlet genexpar where ξ_t is standardnormal

- compute *CAFPE* for all lag combinations up to four lags and highest lag 4 using the quantlet cafpe.



```
"selected lag vector,                estimated CAFPE "
       1        2        0        0   1.4121
"number of lags, chosen lag vector,         estimated CAFPE, plug-in bandwidth"
       0        0        0        0        0   9.9745        0
       1        1        0        0        0    1.519   1.1987
       2        1        2        0        0   1.4121   1.4831
       3        1        2        3        0   2.1042   1.9014
       4        1        2        3        4   7.5668    2.833
```

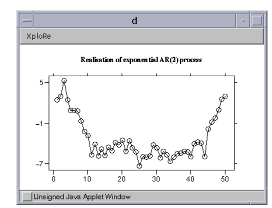
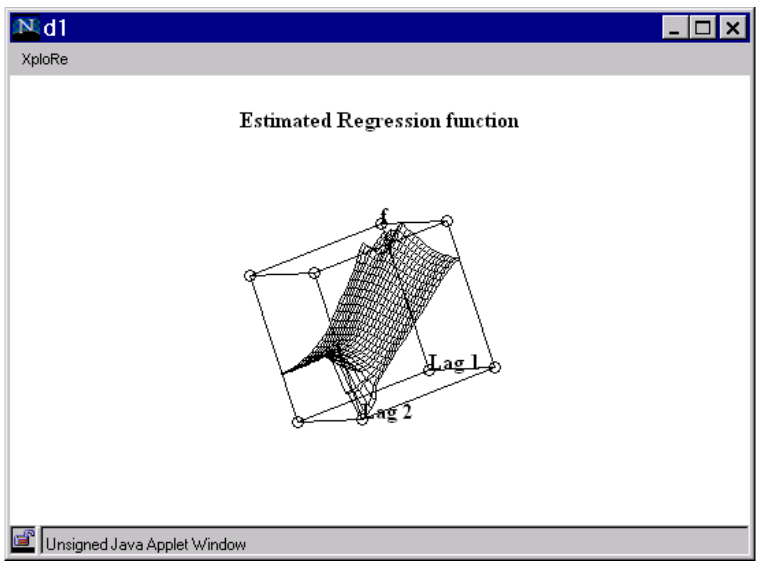


Illustration - function estimation

The plotloclinexample allows to estimate and to plot the conditional function f for the selected lags 1 and 2 and the obtained plug-in bandwidth h_{opt}



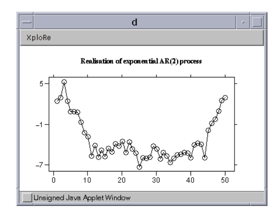
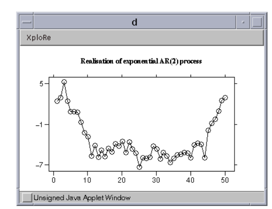


Illustration - advantages of the XQS/XQC architecture

- client's machine is left for client's work
- Example: For $T = 5000$ the example program would take 10.5 hours on a 200 Mhz Pentium in contrast to 2.5 on the Sun Ultra 2 Sparc server.

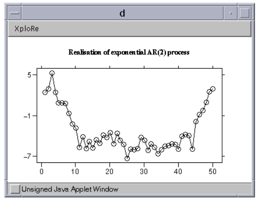


5. ebook

Cooperation with Springer Verlag, Heidelberg

ebook has the following components

- classical printed book
- CD with book and slides
- options for updates
- option for XQS



ebook Statistics of financial markets

- book of 300 pages
- CD with
  - pdf/ps/html version
  - MS Windows XQS
  - Java XQC
  - all necessary Quantlets
  - book update rights ≤ 2 years
  - APSS (auto pilot support system)

(userid: finanz password: mathe)