# Dynamic Combination of Models for Nonlinear Time Series

Peter Bühlmann and Fiorenzo Ferrari
ETH Zürich and BSI SA Lugano
Switzerland

September 2002

**Abstract**

We propose a new method for stationary nonlinear time series analysis which dynamically combines models, either parametric or nonparametric, by using mixture probabilities from so-called variable length Markov chains. The approach is very general and flexible: it can be used for modelling conditional means, conditional variances or conditional densities given the previous lagged values, and the methodology can be applied to dynamically combine almost any kind of models. Parameter estimation (finite or infinite-dimensional) and model selection can be done in a fully data-driven way. We demonstrate the predictive power of the method on finite sample data and an asymptotic consistency result is presented.

Heading: Dynamic combination of models

# 1 Introduction

Nonparametric modeling for stationary nonlinear time series has often been developed by transferring methodology from nonparametric regression to nonparametric autoregression, see for example the overview by Tjøstheim (1994). We adopt here a very different approach which combines flexible Markov modelling for finite spaces with traditional time series models for real-valued data. It can be viewed as a powerful strategy to generalize the finite-valued variable length Markov chains (VLMC), introduced by Rissanen (1983), to real-valued processes. The simplest approach is given by discretizing a real-valued time series which is then modelled with a VLMC (Bühlmann, 1999). The new method here is a substantial improvement over such a discretization scheme and is a hybrid between discretization and "mixture of experts" (Jordan and Jacobs, 1994).

For example, our new modelling approach for the conditional expectation of a stationary time series variable $Y_t$ given its past $\mathcal{F}_{t-1}$, the sigma-algebra generated by the variables $Y_{t-1}, Y_{t-2}, \ldots$, evolves by using a finite-valued variable $X_t \in \{0, 1, \ldots, N-1\}$ and the straightforward identity

$$\mathbb{E}[Y_t|\mathcal{F}_{t-1}] = \sum_{x=0}^{N-1} \mathbb{E}[Y_t|X_t = x, \mathcal{F}_{t-1}]\mathbb{P}[X_t = x|\mathcal{F}_{t-1}]. \tag{1.1}$$

We always construct the variable $X_t$ to depend on $Y_t$ via a discretization scheme, indicating to which discretization interval $Y_t$ belongs to. We can then view the conditional expectation $\mathbb{E}[Y_t|X_t = x, \mathcal{F}_{t-1}]$ as a *local* conditional mean of $Y_t$ given the past and we interpret the probabilities $\mathbb{P}[X_t = x|\mathcal{F}_{t-1}]$ as mixture or combination weights associated to the local conditional means. When specifying a local model for $\mathbb{E}[Y_t|X_t = x, \mathcal{F}_{t-1}]$, for example

$$\mathbb{E}[Y_t|X_t = x, \mathcal{F}_{t-1}] = \mu_x + \sum_{j=1}^{p} \phi_{x,j} Y_{t-j} \tag{1.2}$$

in a local AR($p$) model, we see that (1.1) can be viewed as *dynamic combination of* (local) *models* (DCM), where the word "dynamic" emphasizes that the mixture weights depend on $\mathcal{F}_{t-1}$ and hence (indirectly) on time $t$. We propose to model the mixture probabilities $\mathbb{P}[X_t = x|\mathcal{F}_{t-1}]$ as functions of lagged $X$-variables $\mathbb{P}[X_t = x|X_{t-1}, X_{t-2}, \ldots]$ and model the latter by VLMC's. For choosing a reasonable local model for conditional means, we have (almost) arbitrary flexibility. Particularly, we investigate the local AR model in (1.2) but also show how nonparametric local models can be used. The special model in (1.1) and (1.2) has some connections to autoregressive threshold models in the general form as indicated in Tong and Lim (1980) in their last paragraph on p.285: their thresholding function for different "regimes" could be very general, although most of the implementations are deterministic in terms of lagged values. When choosing in our approach the degenerate case with $X_t = \mathbb{1}_{[Y_{t-d} \le c]}$ for some $d \in \mathbb{N}$ and $c \in \mathbb{R}$, the probabilities $\mathbb{P}[X_t = x|\mathcal{F}_{t-1}]$ take values in $\{0, 1\}$, and our model in (1.1) and (1.2) becomes a self-exciting autoregressive threshold model (Tong, 1990). But usually, our method relies on combining rather than selecting a model whose form depends on the previous observations.

Our methodology for dynamic combination of models can also be used to model the conditional variance given the past $\mathrm{Var}(Y_t|\mathcal{F}_{t-1})$. This, or the volatility as its square root,

are key quantities in econometrics and financial time series. We present in section 5.2 some empirical results about dynamic combination of (local) GARCH models, demonstrating that it often outperforms the popular GARCH(1,1) (Bollerslev, 1986) benchmark model. Further applications of our dynamic combination of models scheme includes estimation of the conditional distribution or density of $Y_t$ given its past $\mathcal{F}_{t-1}$. The methodology is very general and it can be tested out on data whether an (almost) arbitrary model for conditional moments or distributions can be improved via localization and dynamic combination.

# 2 Discretization and variable length Markov chains

We first elaborate how to determine the dynamically changing mixture probabilities $\mathbb{P}[X_t = x|\mathcal{F}_{t-1}]$ in (1.1), assuming stationarity of the observation process $\{Y_t : t \in \mathbb{Z}\}$. The variable $X_t$ takes values in a finite set $\mathcal{X} = \{0, 1, \ldots, N-1\}$ and is constructed from quantile-based discretization of the real-valued observation $Y_t$

$$X_t = q(Y_t),$$
$$q(y) = \begin{cases} 0, & \text{if } -\infty < g(y) \le F^{-1}(\alpha_1) \\ 1, & \text{if } F^{-1}(\alpha_1) < g(y) \le F^{-1}(\alpha_2) \\ \ldots & \\ N-1, & \text{if } F^{-1}(\alpha_{N-1}) < g(y) < \infty \end{cases}, \tag{2.1}$$

where $0 < \alpha_1 < \alpha_2 < \ldots \alpha_{N-1} < 1$, $g(\cdot)$ is a real-valued transformation (not necessarily invertible) and $F$ is the cumulative distribution function of $g(Y_t)$ or its empirical version. For example, with $\alpha_i = i/N$ and using the empirical cumulative distribution of $\{g(Y_t); t = 1, \ldots, n\}$, we have approximately equal number of occurrences of $\{X_t = x\}$ for all $x \in \mathcal{X}$. The most often used transformations are the identity $g(y) = y$ or the quadratic transform $g(y) = y^2$: formula (2.1) then corresponds to discretizing the variables $Y_t$ or $Y_t^2$, respectively.

Stationarity of $\{X_t : t \in \mathbb{Z}\}$ is inherited by $\{Y_t : t \in \mathbb{Z}\}$. Moreover, as the discretization in (2.1) becomes finer, we can approximate $\mathbb{P}[X_t = x|\mathcal{F}_{t-1}]$ in (1.1) by $\mathbb{P}[X_t = x|X_{t-1}, X_{t-1}, \ldots]$, see (Bühlmann, 1999). Variable length Markov chains (VLMC), which we will define in the next section, then yield consistent approximations for any suitably regular stationary process $\{X_t : t \in \mathbb{Z}\}$ (Ferrari and Wyner, 2002) and thus of the mixture weights in (1.1), assuming that the discretization in (2.1) becomes finer.

## 2.1 Variable length Markov chains

Variable length Markov chains (VLMC) have their origin in information theory (Rissanen, 1983) and became recently more established in statistics (Bühlmann and Wyner, 1999).

Denote by $x_i^j = x_j, x_{j-1}, \ldots, x_i$ $(i < j,\ i, j \in \mathbb{Z} \cup \{-\infty, \infty\})$ a vector whose components are written in reverse order, and $wu = (w_{\text{card}(w)}, \ldots, w_2, w_1, u_{\text{card}(u)}, \ldots, u_2, u_1)$ is the concatenation of the vectors $w$ and $u$; here and in the sequel, $\text{card}(\cdot)$ denotes the cardinality of a vector or a set. Also, capital letters $X$ are usually used for random variables and small letters $x$ for deterministic values.

The main idea of a VLMC is that the time-homogeneous transition probabilities

$$\mathbb{P}[X_t = x|X_{t-1}, X_{t-2}, \ldots] = \mathbb{P}[X_t = x|X_{t-1}, \ldots, X_{t-\ell}],\ \ell = \ell(X_{t-1}, X_{t-2}, \ldots),$$

with varying values of $\ell(\cdot)$, depending on the past lagged values $X_{t-1}, X_{t-2}, \ldots$ The formal definition requires more terminology.

**Definition 2.1** *Let $\{X_t : t \in \mathbb{Z}\}$ be a stationary process with values $X_t \in \mathcal{X}$. Denote by $c_{pre} : \mathcal{X}^\infty \to \cup_{j=0}^\infty \mathcal{X}^j \cup \mathcal{X}^\infty$ ($\mathcal{X}^0 = \emptyset$) a (variable projection) function which maps*

$$c_{pre} : x_{-\infty}^0 \mapsto x_{-\ell+1}^0, \text{ where } \ell \text{ is defined by}$$
$$\ell = \ell(x_{-\infty}^0) = \min\{k; \mathbb{P}[X_1 = x_1|X_{-\infty}^0 = x_{-\infty}^0] = \mathbb{P}[X_1 = x_1|X_{-k+1}^0 = x_{-k+1}^0] \text{ for all } x_1 \in \mathcal{X}\},$$
*where $\ell \equiv 0$ corresponds to independence.*

*($\mathbb{P}[X_1 = x_1|X_{-\infty}^0 = x_{-\infty}^0]$ is assumed to be continuous in $x_{-\infty}^0$ with respect to the product topology). The function $c_{pre}(\cdot)$ is called the preliminary context function.*

*Additional structure on the preliminary context function $c_{pre}(\cdot)$ is then built in as follows. The final form of a context function $c(\cdot)$ allows to lump* some *of the values of $c_{pre}(\cdot)$ whose second last symbols are the same (see Example 2 below). Then, $c(\cdot)$ is called a context function and for any $t \in \mathbb{Z}$, $c(x_{-\infty}^{t-1})$ is called the context (the relevant past) at time $t$. In the sequel, a context function $c(\cdot)$ is always meant to be of final form. Let $0 \le p \le \infty$ be the smallest integer such that*

$$_{card}(c(x_{-\infty}^0)) = \ell(x_{-\infty}^0) \le p \text{ for all } x_{-\infty}^0 \in \mathcal{X}^\infty.$$

*The number $p$ is called the order of the context function $c(\cdot)$, and if $p < \infty$, $\{X_t : t \in \mathbb{Z}\}$ is called a stationary variable length Markov chain [VLMC] of order $p$.*

Due to stationarity of $\{X_t : t \in \mathbb{Z}\}$, transition probabilities are homogeneous in time and the restriction to indices $0, -1, \ldots$ in the definitions above is without loss of generality. Clearly, a VLMC of order $p$ is a Markov chain of order $p$, with the additional structure of having a *memory of variable length $\ell$*. If the context function $c(\cdot)$ of order $p$ is the full projection $x_{-\infty}^0 \mapsto x_{-p+1}^0$ for all $x_{-\infty}^0$, the VLMC is a full Markov chain of order $p$. A VLMC has an important representation as a graphical tree model, see Figure 1.

**Definition 2.2** *Let $c(\cdot)$ be a context function of a stationary VLMC. The context tree $\tau$ is defined as*

$$\tau = \tau_c = \{w; w = c(x_{-\infty}^0),\ x_{-\infty}^0 \in \mathcal{X}^\infty\}.$$

The context function $c(\cdot)$ can be reconstructed from $\tau_c$. The context tree $\tau_c$, which does not have to be complete with $\text{card}(\mathcal{X})$ offsprings per internal node, is nothing else than the minimal state space of the VLMC.

*Example 1.* $\mathcal{X} = \{0, 1\}$, order $p = 3$.
The function

$$c(x_{-\infty}^0) = \begin{cases} 0, & \text{if } x_0 = 0,\ x_{-\infty}^{-1} \text{ arbitrary} \\ 1,0,0, & \text{if } x_0 = 1, x_{-1} = 0, x_{-2} = 0,\ x_{-\infty}^{-3} \text{ arbitrary} \\ 1,0,1, & \text{if } x_0 = 1, x_{-1} = 0, x_{-2} = 1,\ x_{-\infty}^{-3} \text{ arbitrary} \\ 1,1, & \text{if } x_0 = 1, x_{-1} = 1,\ x_{-\infty}^{-2} \text{ arbitrary} \end{cases}$$

can be represented by the tree (read top down) $\tau_c = \{0, 100, 101, 11\}$ on the left hand side in Figure 1.

*Example 2.* $\mathcal{X} = \{0, 1, 2, 3\}$, order $p = 2$.
Consider first the preliminary context function

$$c_{pre}(x_{-\infty}^{0}) = \begin{cases} 0 & \text{if } x_0 = 0,\ x_{-\infty}^{-1} \text{ arbitrary} \\ 1 & \text{if } x_0 = 1,\ x_{-\infty}^{-1} \text{ arbitrary} \\ 2 & \text{if } x_0 = 2,\ x_{-\infty}^{-1} \text{ arbitrary} \\ 3,0 & \text{if } x_0 = 3,\ x_{-1} = 0,\ x_{-\infty}^{-2} \text{ arbitrary} \\ 3,1 & \text{if } x_0 = 3,\ x_{-1} = 1,\ x_{-\infty}^{-2} \text{ arbitrary} \\ 3,2 & \text{if } x_0 = 3,\ x_{-1} = 2,\ x_{-\infty}^{-2} \text{ arbitrary} \\ 3,3 & \text{if } x_0 = 3, x_{-1} = 3,\ x_{-\infty}^{-2} \text{ arbitrary} \end{cases}$$

Assume now the additional structure that the contexts $30, 31, 32$ are to be viewed as the same and lumped to $3[0, 1, 2]$, where $[\cdot]$ denotes regular expression. The (final form) context function is then

$$c(x_{-\infty}^{0}) = \begin{cases} 0 & \text{if } x_0 = 0,\ x_{-\infty}^{-1} \text{ arbitrary} \\ 1 & \text{if } x_0 = 1,\ x_{-\infty}^{-1} \text{ arbitrary} \\ 2 & \text{if } x_0 = 2,\ x_{-\infty}^{-1} \text{ arbitrary} \\ 3,[0,1,2] & \text{if } x_0 = 3,\ x_{-1} \in \{0,1,2\},\ x_{-\infty}^{-2} \text{ arbitrary} \\ 3,3 & \text{if } x_0 = 3, x_{-1} = 3,\ x_{-\infty}^{-2} \text{ arbitrary} \end{cases}$$

This context function can be represented by the tree (read top down) $\tau_c = \{0, 1, 2, 3[0, 1, 2], 33\}$ on the upper right hand side in Figure 1. An alternative tree representation, which is notationally and algorithmically more efficient and which we will use from now on, is given by the tree $\tau_c = \{0, 1, 2, 3, 33\}$ on the lower right hand side in Figure 1, where the context $3[0, 1, 2]$ is represented by the *internal* node 3.

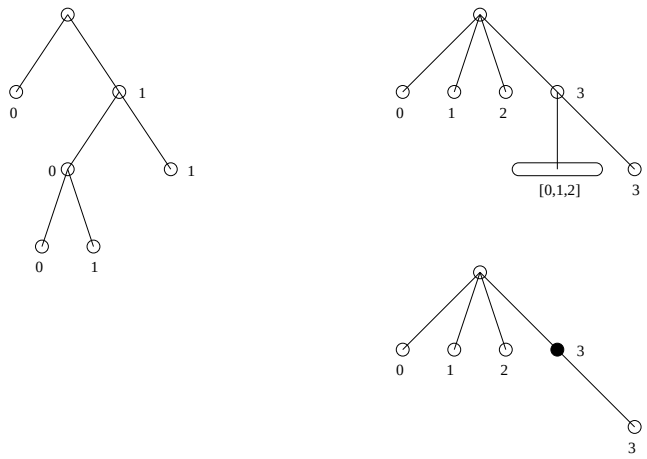


Figure 1: Tree representations of the variable-length memories in Examples 1 and 2.

## 2.2 The context algorithm for fitting VLMC's

Fitting a VLMC involves a version of the tree structured context algorithm (Rissanen, 1983) for estimating the variable length memory, described by the context function $c(\cdot)$, and for the set of transition probabilities. Estimation of $\tau_c$ is a highly complex model selection problem; due to the extremely large number of possible models, a natural tree hierarchy is employed.

One tuning parameter needs to be specified, the so-called cutoff denoted by $K$. Asymptotically, the cutoff should be of the form $K = K_n \sim C \log(n)$ with $C > 2\text{card}(\mathcal{X}) + 3$ a suitable constant. The choice of the cutoff parameter $K = K_n$ in practice is often more intuitive on the scale of $\chi^2$-quantiles, see Bühlmann and Wyner (1999):

$$K = \chi^2_{N-1;Q}/2, \text{ half of the } Q\text{th quantile of a } \chi^2_{N-1} \text{ distribution,} \tag{2.2}$$

where $Q$ has to be chosen.

The algorithm yields then estimates $\hat{c}(\cdot)$ (or $\hat{\tau}$) of the context function (context tree) and estimated transition probabilities $\hat{\mathbb{P}}[X_t = x|\hat{c}(X_1^{t-1})]$ for the target $\mathbb{P}[X_t = x|c(X_{-\infty}^{t-1})]$. These estimates are consistent for the true context tree and transition probabilities (and also for marginal distributions) of suitably regular, stationary $\mathcal{X}$-valued processes which do not necessarily need to be a VLMC, see Bühlmann and Wyner (1999) and Ferrari and Wyner (2002).

A detailed description of the algorithm is given in Appendix A. The context algorithm is implemented in the statistical computing language *R*, freely available from the download section of http://www.r-project.org/. On-line help is available in *R* with library(VLMC). A more detailed tutorial for fitting and modelling with VLMC's is given in Mächler and Bühlmann (2002).

## 2.3 Quantized variable length Markov chains

Quantized variable length Markov chains are real-valued processes $\{Y_t : t \in \mathbb{Z}\}$ which evolve rather directly from variable length Markov chains, see Bühlmann (1999). Their conditional densities, assuming they exist, are of the form

$$f_{Y_t|\mathcal{F}_{t-1}}(y|\mathcal{F}_{t-1}) = \sum_{x=0}^{N-1} f_x(y)\mathbb{P}[X_t = x|c(X_{-\infty}^{t-1})],$$

where $f_x(\cdot)$ are univariate densities. The conditional expectation for such models is then of the form

$$\mathbb{E}[Y_t|\mathcal{F}_{t-1}] = \sum_{x=0}^{N-1} \theta_x \mathbb{P}[X_t = x|c(X_{-\infty}^{t-1})],$$

where $\theta_x = \int y f_x(y) dy$ can be interpreted as a quantization value corresponding to the discretized $x$. Comparing this with (1.1), we see that this is a dynamic combination of constants. Our new methodology will be much more flexible by allowing more complex (local) models instead of just constants.

# 3 Dynamic combination: models and estimation

Dynamic combination of models can be set-up in a wide variety of settings such as for conditional means, variance or distributions. In the sequel, we abbreviate by $P_{t,x} = \mathbb{P}[X_t = x|c(X_{-\infty}^{t-1})]$ the transition probabilities in a VLMC and by $\hat{P}_{t,x} = \hat{\mathbb{P}}[X_t = x|\hat{c}(X_1^{t-1})]$ its estimates from the context algorithm.

## 3.1 Examples of models

### 3.1.1 Conditional mean models

Consider the general homoscedastic error model

$$Y_t = \mu_t + \sigma\varepsilon_t, \tag{3.1}$$

where $\{\varepsilon_t; t \in \mathbb{Z}\}$ is an i.i.d. innovation sequence with $\mathbb{E}[\varepsilon_t] = 0, \mathrm{Var}(\varepsilon_t) = 1$ and $\varepsilon_t$ independent from $\{Y_s : s < t\}$. The conditional mean is modelled with DCM

$$\mu_t = \sum_{x=0}^{N-1} m_x(\mathcal{F}_{t-1})P_{t,x}$$

with local models $m_x(\mathcal{F}_{t-1})$. For example, they can be parametric autoregressive,

$$m_x(\mathcal{F}_{t-1}) = m(\theta_x, Y_{t-p}^{t-1}) = \phi_{x,0} + \sum_{j=1}^{p} \phi_{x,j}Y_{t-j},\ \theta_x = (\phi_{x,0}, \dots \phi_{x,p})', \tag{3.2}$$

or they can be additive of order $p$

$$m_x(\mathcal{F}_{t-1}) = m_x(Y_{t-p}^{t-1}) = \alpha_x + \sum_{j=1}^{p} f_{x,j}(Y_{t-j}), \tag{3.3}$$

with identifiability constraints $\mathbb{E}[f_{x,j}(Y_1)] = 0$ for all $j$ and $x$.

### 3.1.2 Conditional variance models

Consider the stochastically changing heteroscedastic error model

$$Y_t = \mu_t(\gamma) + \sigma_t\varepsilon_t, \tag{3.4}$$

where $\varepsilon_t$ are as in (3.1) and for simplicity, we assume that $\mu_t(\gamma) = \mathbb{E}[Y_t|\mathcal{F}_{t-1}]$ is of simple parametric form such as $\mu_t(\gamma) = \gamma Y_{t-1}$. The conditional variance $\sigma_t^2$ given $\mathcal{F}_{t-1}$ is modelled by DCM, for example with local GARCH(1,1) models,

$$\begin{aligned} \sigma_t^2 &= \sum_{x=0}^{N-1} v_x(\mathcal{F}_{t-1})P_{t,x}, \\ v_x(\mathcal{F}_{t-1}) &= \alpha_{x,0} + \alpha_{x,1}Y_{t-1}^2 + \beta_x\sigma_{t-1}^2, \end{aligned} \tag{3.5}$$

where $\alpha_{x,0}, \alpha_{x,1}, \beta_x \geq 0$ for all $x$.

## 3.2 Estimation

We describe now parameter or curve estimation in the homoscedastic and heteroscedastic models (3.1) and (3.4), respectively. For a given discretizer $q(\cdot)$ and a given structure of the local model, we use maximum likelihood for finite-dimensional parameter estimation. For the nonparametric DCM in (3.3), a weighted backfitting method is proposed.

### 3.2.1 Conditional mean: local parametric models and Gaussian DC-AR

For the homoscedastic model (3.1), we focus for expository simplicity to the case where the local model is parametric and Markovian of order $p$ (the non-Markovian case is analogous to the heteroscedastic model described in section 3.2.2). The conditional mean can then be parameterized as

$$\mu_t(\theta) = \sum_{x=0}^{N-1} m(\theta_x; Y_{t-p}^{t-1}) P_{t,x}, \ \theta = (\theta_0, \ldots, \theta_{N-1})'. \tag{3.6}$$

see also (3.2).

We denote by

$$\hat{\mu}_t(\theta) = \sum_{x=0}^{N-1} m(\theta_x; Y_{t-p}^{t-1}) \hat{P}_{t,x}, \tag{3.7}$$

where $\hat{P}_{t,x}$ is the estimate from the context algorithm. The conditional log-likelihood, given the first $s = \max\{p, d\}$ observations $Y_1^s$, with $d$ the order of the VLMC, is then

$$\sum_{t=s+1}^{n} \log(\frac{1}{\sigma} f_\varepsilon(\frac{Y_t - \hat{\mu}_t(\theta)}{\sigma})),$$

where $f_\varepsilon(\cdot)$ denotes the density of the innovation $\varepsilon_t$. Now, replacing the variance $\sigma^2$ with the estimate $\hat{\sigma}^2(\theta) = \sum_{t=s+1}^{n} (Y_t - \hat{\mu}_t(\theta))^2/(n-s)$ yields for the conditional log-likelihood

$$\ell_n(\theta) = -(n-s)\log(\hat{\sigma}^2(\theta)) + \sum_{t=s+1}^{n} \log\left(f_\varepsilon(\hat{\sigma}^{-1}(\theta)(Y_t - \hat{\mu}_t(\theta)))\right). \tag{3.8}$$

The maximum-likelihood estimator is $\hat{\theta}_{MLE} = \mathrm{argmin}_\theta - \ell_n(\theta)$.

In case of standard Gaussian innovations $\varepsilon_t$, the maximum likelihood estimator coincides with least squares

$$\hat{\theta}_{LS} = \mathrm{argmin}_\theta \sum_{t=s+1}^{n} (Y_t - \hat{\mu}_t(\theta))^2. \tag{3.9}$$

Furthermore, if the local model is autoregressive as in (3.2), the least squares estimator is linear and can be solved explicitly. To make the model identifiable we use instead of $\hat{P}_{t,N-1}$ the equivalent quantity $1 - \sum_{x=0}^{N-2} \hat{P}_{t,x}$. This induces a re-parameterization: the (old) parameter vector $\theta = (\theta_0, \ldots, \theta_{N-1})$ in (3.2) becomes in the new representation

$$\begin{aligned} &\theta' = (\theta_0', \theta_1', \ldots, \theta_{N-1}'), \\ &\theta_x' = (\phi_{x,N-1}, \phi_{x,0} - \phi_{x,N-1}, \phi_{x,1} - \phi_{x,N-1}, \ldots, \phi_{x,N-2} - \phi_{x,N-1}). \end{aligned} \tag{3.10}$$

The least squares estimate of $\theta'$ of dimension $N(p+1)$ is then

$$\hat{\theta}'_{LS,n} = (A^T A)^{-1} A^T Y_{s+1}^n, \tag{3.11}$$

where $A$ is an $(n-s) \times N(p+1)$ matrix with full rank $N(p+1)$ whose exact form is given in Appendix B. This estimate $\hat{\theta}'_{LS}$ can be efficiently computed via the QR decomposition.

### 3.2.2 Conditional variance: DC-GARCH models

For the stochastically heteroscedastic model, we focus on the local GARCH(1,1) model in (3.5) as an interesting example. The conditional log-likelihood in the model (3.4)-(3.5), given the first $s$ observations and an initial conditional variance $\sigma_s^2$, then becomes

$$\ell_n(\theta;\gamma) = \sum_{t=s+1}^{n} \log\left(\frac{1}{\sigma_t(\theta)} f_\varepsilon\left(\frac{Y_t - \mu_t(\gamma)}{\sigma_t(\theta)}\right)\right)$$

where $\theta = \{\alpha_{x,0}, \alpha_{x,1}, \beta_x; x = 0, \ldots, N-1\}$ whose components are positive. The maximum-likelihood estimator is defined as $(\hat{\theta}, \hat{\gamma})_{MLE} = \text{argmin}_{\theta,\gamma} - \ell_n(\theta, \gamma)$.

### 3.2.3 Weighted backfitting for nonparametric local mean models

We consider now conditional mean models (3.1), where the local models $m_x(\cdot)$ are non-parametric but additive as in (3.3). When focusing on least squares, a weighted backfitting can be used to estimate the local functions $m_x(\cdot)$.

Our weighted backfitting algorithm then proceeds as follows:

Step 1 Initialize the functions $\hat{m}_x^{(0)}(\cdot) \equiv n^{-1} \sum_{t=1}^n Y_t$.

Step 2 Cycle through the components k=0,1,...,N-1,0,1,...

$$\hat{m}_k(\cdot) \text{ minimizes over } m_k = \alpha_k + \sum_{j=1}^{p} f_{k,j}(\cdot) \text{ the criterion:}$$

$$\sum_{t=s+1}^{n} (Y_t - \sum_{x=0, x\neq k}^{N-1} \hat{m}_x(Y_{t-p}^{t-1}) \hat{P}_{t,x} - m_k(Y_{t-p}^{t-1}) \hat{P}_{t,k})^2$$

$$= \sum_{t=s+1}^{n} \hat{P}_{t,k}^2 (R_{t,k} - m_k(Y_{t-p}^{t-1}))^2, \; R_{t,k} = (Y_t - \sum_{x=0, x\neq k}^{N-1} \hat{m}_x(Y_{t-p}^{t-1}))/\hat{P}_{t,k},$$

where the minimization respects smoothness constraints. The method requires that the additive function estimation routine for $\hat{m}_k$ can be used in a version respecting weights $\hat{P}_{t,k}^2$.

Step 3 Continue until the relative change of the least squares criterion falls below a given tolerance such as $10^{-6}$.

In case of local additive models as in (3.3), we use smoothing splines for every component $f_{x,j}(\cdot)$ but other smoothers could be used as well.

### 3.3 Theoretical properties for local autoregressions

We show here for the DC-AR model in (3.2), that the least squares estimator in (3.9) (or in (3.11)) converges to a unique $\theta_*$.

**Theorem 3.1** *Under the assumptions (B1)-(B3) and (C) described in Appendix B, the least squares estimator (3.11) for the DC-AR(p) model is consistent:*

$$\hat{\theta}'_{LS,n} = \theta'_* + o_P(1) \ (n \to \infty),$$

*where $\theta'_*$ is the unique minimizer of $\mathbb{E}[(Y_t - \mu_t(\theta'))^2]$ with respect to $\theta'$, and where $\mu_t(\theta')$ is specified by (3.2) but with the parameterization in (3.10).*

A proof is given in Appendix B. Theorem 3.1 establishes consistency for the best parameter $\theta_*$ in a DC-AR model with specified quantizer and order of the local AR-models: the specified DC-AR model with parameter $\theta_*$ is the closest to the true data-generating process with respect to the Kullback-Leibler divergence, among all such DC-AR's with other parameters $\theta$. If the data-generating process is of the specified DC-AR model form with parameter $\theta_0$, Theorem 3.1 then establishes convergence to the true $\theta_0$. Due to convexity of the least squares optimization problem, the parameter $\theta^*$ and the least squares estimator $\hat{\theta}_{LS}$ are unique, a property which typically does not hold for complex nonparametric procedures such as projection pursuit (auto-) regression (Friedman and Stuetzle, 1981). The convergence rate of $\hat{\theta}_{LS}$ is typically slower than $1/\sqrt{n}$, because the generally infinite-dimensional mixture probabilities $P_{t,x}$ have to be estimated.

Already the quantized variable length Markov chains from section 2.3 have been shown to be dense in the set of stationary processes (Bühlmann, 1999). Essentially under the conditions from Theorem 3.1, estimated quantized variable length Markov chains consistently approximate all finite-dimensional distributions of the data-generating process (Ferrari and Wyner, 2002); and the same can be shown for estimated DC-AR or estimated dynamic combination of many other models.

## 4 Choice of discretization and model selection

Choosing a discretizer as in (2.1) and selecting a structure in a class of local models, for example the order $p$ in local autoregressions in (1.2), amounts to a discrete optimization problem and typically cannot be solved by exhaustive search. Two search strategies among the possible discretizers are proposed which are then combined with local model selection. As an overall criterion to be minimized, we focus on penalized likelihoods.

For conditional mean models, we always choose the discretizer in (2.1) with the identity $g(y) = y$ while for conditional variance models, we always use $g(y) = y^2$ reflecting the second moment structure of a variance.

### 4.1 Balanced and recursive discretizers

The balanced discretizer is as in (2.1) with $\alpha_i = i/N$ and $F$ the empirical cumulative distribution function of $\{g(Y_t); t = 1, \ldots, n\}$, resulting in approximately equal number of occurrences of $\{X_t = x\}$. The best balanced discretizer is thus specified by the optimal

number $N$ of discretization values. Balanced discretizers work well for conditional mean models and if the data do not exhibit any special ranges in $\mathbb{R}$, such as negative or positive extremes, which would affect the dynamics of the process very much.

There are noticeable cases, where extreme events strongly influence the dynamics of the underlying process: a prime example are time series of returns from financial instruments, see also section 5.2. We then propose a recursive discretization scheme which works as follows.

*Step 1 (initialization):* Start with no discretization, corresponding to $N = 1$. Set $\mathcal{A} = \mathbb{R}$ and $m = 0$.

*Step 2:* Refine the selected partition interval from $\mathcal{A}$, say $\mathcal{R}^*$ into two disjoint intervals $\mathcal{R}^*_{left}, \mathcal{R}^*_{right}$ ($\mathcal{R}^* = \mathcal{R}^*_{left} \cup \mathcal{R}^*_{right}$) such that the total negative log-likelihood (requires estimation of parameters in local model) is minimal. This refinement is performed on a grid of quantile values of the empirical cumulative distribution function of $\{g(Y_t); t = 1, \ldots, n\}$. Set $\mathcal{A} = \mathcal{A} \setminus \mathcal{R}^* \cup \{\mathcal{R}^*_{left}, \mathcal{R}^*_{right}\}$ and increase m by one.

*Step 3:* Repeat Step 2 until $m$ reaches a pre-specified level $M$.

This method is a recursive binary tree-structured search for $N = M + 1$ partition intervals or equivalently, $N - 1 = M$ quantiles in the discretizer (2.1); and again, only $N$ needs to be selected. Note that we implicitly assume here that a local model structure is given for evaluating the log-likelihood.

### 4.2 Model selection

Having chosen the class of local models and the function $g(\cdot)$ in the discretization scheme (2.1) we need to choose the amount of discretization and the dimensionality of the local model. We propose to minimize the AIC statistic

$$AIC = -2 \text{ log-likelihood} + 2(N \dim(\text{local model}) + (N-1)\text{card}(\hat{\tau})),$$

where $\hat{\tau}$ is the estimated state space of the VLMC for the dynamic mixture weights. Minimization is either over a balanced or recursive discretizer as described in section 4.1 and typically with a hierarchy over local models such as the order of a local AR model.

Of course, estimation of the VLMC involves the cut-off $K$ as a further tuning parameter. We can also incorporate this into the search with AIC: then, the log-likelihood as well as $\hat{\tau}$ depend also on $K$.

## 5 Numerical examples

### 5.1 Simulations for estimating conditional means

The main objective here is to compare the DC-AR model with the parametric AR model (AR), with projection pursuit (PPR) (Friedman and Stuetzle, 1981) and with nonparametric additive modelling (AM) (cf. Hastie and Tibshirani, 1990), both of the latter using lagged values as predictors.

We simulate data from the following nonlinear model:

$$Y_t = \mu_t + \sigma\varepsilon_t,$$

$$\mu_t = [1.05 - 2.15\cos(\pi Y_{t-1})\exp(-0.5Y_{t-1}^2)]Y_{t-1}$$
$$-[0.15 - 0.90\sin(\pi Y_{t-2})\exp(-0.5(Y_{t-1}^2 + Y_{t-2}^2)]Y_{t-2}$$
$$-[0.55 - 1.60\exp(-0.5(Y_{t-1}^2 + \mu_{t-1}^2)]\mu_{t-1}, \qquad (5.1)$$

where $\{\varepsilon_t; t \in \mathbb{Z}\}$ is an i.i.d. innovation sequence, $\varepsilon_t \sim \mathcal{N}(0,1)$ independent from $\{Y_s; s < t\}$, and $\sigma^2 = 0.5, 1, 2$ or 4. Note that this model is non-Markovian while all the fitting techniques yield Markovian models for $\mu_t$. We choose (training) sample size n=2000 and generate a test set of size 2000 with the subsequent values $Y_{n+1}, \ldots, Y_{2n}$. We always simulate over 100 realizations from model (5.1).

For fitting DC-AR models, we use model selection as in section 4.2 for the balanced discretizer in section 4.1 and for the local model order which was forced to be the same for all local models. Also, the cut-off parameter $K$ in the Context Algorithm has been set equal to $\chi^2_{N-1;0.95}/2$, see also (2.2), which is a reasonable ad-hoc value. For all other models, namely AR, PPR and AM, the selection was also done by minimizing the AIC statistic, using the equivalent degrees of freedom for the nonparametric methods.

The resulting outsample mean squared prediction errors $2000^{-1}\sum_{t=2001}^{4000}(Y_t - \hat{\mu}_t)^2$ are summarized in Figure 2. Overall, additive modelling has a slight edge over DC-AR,

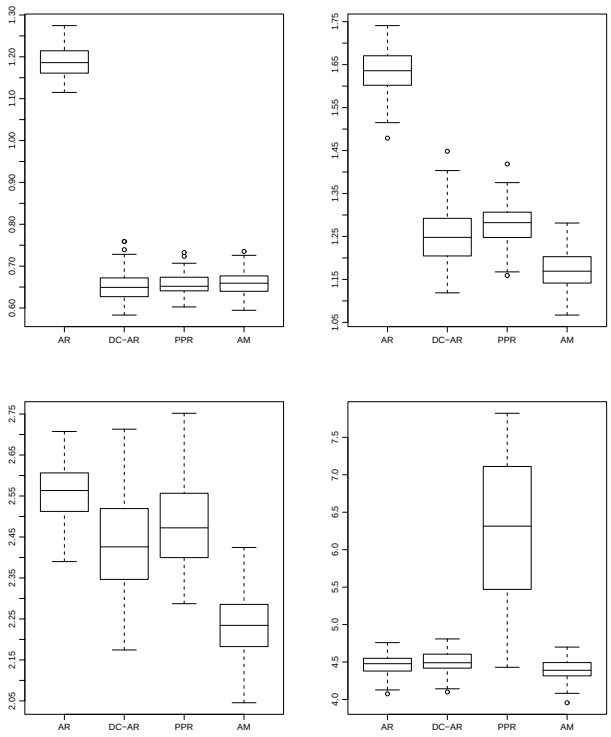


Figure 2: Mean squared prediction errors in model (5.1). In lexicographic order: $\sigma_t^2 = 0.5, 1, 2, 4$.

and DC-AR is a bit better than PPR; the parametric AR is not competitive for the nonlinear model (5.1) except when the innovation variance is large. Similar findings, described in detail in Ferrari (2002), have been obtained when looking at $t_\nu$-distributed

innovations ($\nu = 4, 6, 8$) or stochastically changing variances $\sigma^2 = \sigma_t^2$ in model (5.1); also, the results were similar when looking at the mean absolute prediction error. Although additive modelling performed here best, we show in section 5.1.2 that we can sometimes improve it further by using dynamic combination of additive models as described in section 3.2.3. Thus, as mentioned before, the potential of DCM is not restricted to local AR models, although the simple DC-AR was found to perform well in model (5.1) and in a variety of other processes

### 5.1.1 Computational efficiency

To give a crude idea about computational efficiency, Table 1 shows the CPU times needed to compute the mean squared prediction errors for the 100 model simulations in (5.1), including all elements of tuning and model selection. For AR, PPR and AM, we use the implementation in R (http://www.r-project.org) and the results are based on a Linux machine with Pentium III processor, 930MHz and 256 MB RAM. Fitting DC-AR models was 2-5 times faster than AM and much faster than PPR. The same applies with other datasets. Note that fitting the parameters in a DC-AR amounts to estimating a VLMC and a convex optimization problem which has a unique and explicite solution.

| | $\sigma^2 = 0.5$ | $\sigma^2 = 1$ | $\sigma^2 = 2$ | $\sigma^2 = 4$ |
|---|---|---|---|---|
| AR | 587 | 589 | 591 | 594 |
| DC-AR | 2668 | 2663 | 3450 | 1664 |
| PPR | 31678 | 25278 | 18057 | 448030 |
| AM | 6040 | 8293 | 8240 | 8175 |

Table 1: CPU times for model (5.1).

Thus, the DC-AR models work similarly well as PPR (a little better) and as AM (a little worse), see section 5.1, but are fitted remarkably faster than these other two nonparametric methods.

### 5.1.2 Dynamic combination of nonparametric additive models

We exemplify here how to improve an additive model by dynamic combination with a recursive discretizer as in section 4.1 with $N = 2$. Consider a nonlinear random coefficient model

$$\begin{aligned} Y_t &= U_{1,t} Y_{t-1} + U_{2,t} Y_{t-2} + \sigma_t \varepsilon_t, \\ \sigma_t^2 &= 0.1 + 0.25 Y_{t-1}^2, \end{aligned} \tag{5.2}$$

where $\{U_{1,t} : t \in \mathbb{Z}\}$ and $\{U_{2,t} : t \in \mathbb{Z}\}$ are independent i.i.d. sequences, uniformly distributed on $(-1, 1)$ and $\varepsilon_t$ as in (5.1). As before, (training) sample size is chosen as $n = 2000$ and 2000 subsequent values serve as a test sample.

The strategy employed here for model fitting is to first select (and estimate) a AM using AIC for order selection; we then try to improve it by estimating a DC-AM of the same (local) order for $N = 2$ quantization intervals. The value $\alpha_1 = 0.9$ in (2.1) turned

out to be a good discretizer and the cut-off parameter of the context algorithm is again used with the "default" value $K = \chi^2_{N-1;0.95}/2$.

Figure 3 displays the mean squared prediction error for 100 simulations from (5.2). We find for this model, that DC-AM also yields an improvement over an optimally AIC-tuned AM.

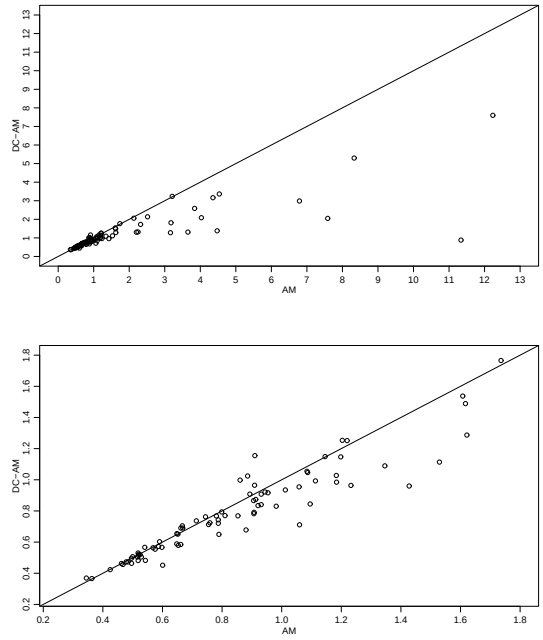


Figure 3: Top: Mean squared prediction error of DC-AM versus mean squared prediction error of AM for 100 simulations from model (5.2). Bottom: Zoom of top panel.

## 5.2 Volatility estimation for daily stock returns

The value of DCM is not primarily for homoscedastic error models where quite many standard nonparametric techniques work reasonably well. When it comes to estimating the important conditional variance for financial time series in multiplicative models, log-transformed nonparametric additive modelling has been found to predict rather poorly (Audrino and Bühlmann, 2001), and DCM offers here an interesting way to improve the parametric ARCH or GARCH models.

Consider the daily log-returns $Y_t = \log(P_t/P_{t-1})$ of the prices $P_t$ of a financial asset. The volatility is defined as $\sqrt{\mathrm{Var}(Y_t|\mathcal{F}_{t-1})}$, which is the conditional standard deviation given the information up to time $t-1$. For such financial time series of log-returns, the conditional expectation is not of primary importance and we consider the AR(1)-DC-GARCH(1,1) model

$$Y_t = \gamma Y_{t-1} + \sigma_t(\theta)\varepsilon_t \tag{5.3}$$

where $\sigma_t^2(\theta)$ is specified in (3.5). For the innovation process $\{\varepsilon_t : t \in \mathbb{Z}\}$ we assume either

of the following distributions

$$\varepsilon_t \sim \mathcal{N}(0,1)\ ,$$
$$\varepsilon_t = \sqrt{(\nu-2)/\nu} Z_t\,, \quad Z_t \sim t_\nu\,.$$

Estimation of parameters is done as described in section 3.2.2. In case of scaled $t_\nu$ distributed innovations, $\nu$ is treated as one additional unknown parameter in the log-likelihood. The discretization and model selection is done with the recursive discretizer and AIC as described in sections 4.1 and 4.2. The context algorithm is used with fixed cut-off tuning parameter $K = \chi^2_{N-1;0.975}/2$, see (2.2): this is a relatively large value of $K$ to favor models which are not too far away from the GARCH(1,1) since $K$ sufficiently large always yields an estimated VLMC of order zero.

We measure the goodness of fit by evaluating the outsample negative log-likelihood (OS-NLL) on a test sample. The training data $Y_1, \ldots, Y_n$ is used for estimation only and a test sample $Y_1^*, \ldots, Y_{n_{test}}^*$ is used for evaluating

$$\text{OS-NLL} = -\ell_n(\hat\theta, \hat\gamma; Y_1^*, \ldots Y_{n_{test}}^*) = -\sum_{t=s+1}^{n_{test}} \log\left(\frac{1}{\sigma_t(\hat\theta)} f_\varepsilon\left(\frac{Y_t - \hat\gamma Y_{t-1}}{\sigma_t(\hat\theta)}\right)\right)$$

where $f_\varepsilon(\cdot)$ is either the density of a standard normal or of a scaled $t_\nu$ distribution. In the latter case, the evaluated negative log-likelihood involves also the estimated degrees of freedom $\nu$. Others measures are the in- and outsample squared prediction errors

$$\text{IS-L2-PE} \;=\; \frac{1}{n-s}\sum_{t=s+1}^{n}\left(\sigma_t(\widehat\theta)^2 - (Y_t - \widehat\gamma Y_{t-1})^2\right)^2\,,$$
$$\text{OS-L2-PE} \;=\; \frac{1}{n^*-s}\sum_{t=s+1}^{n^*_{test}}\left(\sigma_t^*(\widehat\theta)^2 - (Y_t^* - \widehat\gamma Y_{t-1}^*)^2\right)^2\,,$$

where $\sigma_t$ and $\sigma_t^*$ denote the volatilities evaluated at the in- and outsample observations, respectively.

### 5.2.1 The BMW stock

We consider daily negative log-returns (in percentage) of the BMW stock price from November 23, 1988 to July 23, 1996, which correspond to a sample of 2000 data points. The former half is the training set, the latter for model testing.

Our fitting procedure selects and estimates a AR(1)-DC-GARCH(1,1) model in (5.3) with $N = 2$ quantization intervals determined by $\alpha = 0.4$ in (2.1), both for normal and $t$-distributed innovations. Detailed results are given in Table 2, comparing also with an AR(1)-GARCH(1,1) model. The gain with respect to the AIC statistic is clearly visible, mostly for normal innovations and the improvements in terms of the outsample squared prediction error are substantial. To appreciate this, we point out that with real financial data, differences in the goodness of fit measures are often heavily masked due to high noise, cf. Audrino and Bühlmann (2001); for example, for the OS-L2-PE, we replace the target $\sigma_t^{*2}$, the true squared volatility, by the very noisy estimate $(Y_t^* - \hat\gamma Y_{t-1}^*)^2$. Table 3 displays the predictive performance for AR(1)-DC-GARCH(1,1) models on $N = 2$ discretization

| | GARCH(1,1) | DC-GARCH(1,1) | gain |
|---|---|---|---|
| AIC | 3566.4 | 3536.4 | 30 |
| IS-L2-PE | 76.472 | 74.959 | 2% |
| OS-NLL | 1.607 | 1.607 | 0% |
| OS-L2-PE | 12.836 | 11.064 | 16% |
| N | – | 2 | – |
| $\alpha$ | – | 0.4 | – |

| | GARCH(1,1) | DC-GARCH(1,1) | gain |
|---|---|---|---|
| AIC | 3339.2 | 3335.9 | 3.3 |
| IS-L2-PE | 76.079 | 75.912 | 0.2% |
| OS-NLL | 1.546 | 1.538 | 0.52% |
| OS-L2-PE | 12.580 | 11.266 | 12% |
| $\widehat{\nu}$ | 3.75 | 3.79 | – |
| N | – | 2 | – |
| $\alpha$ | – | 0.4 | – |

Table 2: BMW data. Top: results for normal innovations. Bottom: results for $t_\nu$ innovations.

| AR(1)-DC-GARCH(1,1), N = 2,N(0,1) | | | AR(1)-DC-GARCH(1,1), N = 2, $t_\nu$ | | |
|---|---|---|---|---|---|
| $\alpha$ | OS-NLL | OS-L2-PE | $\alpha$ | OS-NLL | OS-L2-PE |
| 0.1 | 1.608 | 12.335 | 0.1 | 1.540 | 11.667 |
| 0.2 | 1.658 | 11.219 | 0.2 | 1.550 | 11.530 |
| 0.3 | 1.812 | 11.676 | 0.3 | 1.602 | 12.555 |
| 0.4 ("AIC") | 1.607 | 11.064 | 0.4 ("AIC") | 1.538 | 11.266 |
| 0.5 | 1.759 | 11.798 | 0.5 | 1.586 | 12.118 |
| 0.6 | 1.659 | 11.383 | 0.6 | 1.561 | 11.630 |
| 0.7 | 1.608 | 12.058 | 0.7 | 1.548 | 11.756 |
| 0.8 | 1.608 | 11.296 | 0.8 | 1.548 | 11.672 |
| 0.9 | 1.611 | 12.047 | 0.9 | 1.542 | 11.632 |

Table 3: BMW data. Outsample negative log-likelihood and mean squared prediction error. Left: normal innovations. Right: $t_\nu$ innovations. Model selected by AIC is denoted by "AIC".

intervals with quantiles $\alpha \in \{0.1, 0.2, \ldots, 0.9\}$. The model selected by AIC in the AR(1)-DC-GARCH(1,1) class, denoted by "AIC", is the optimal model in the sense that its OS-NLL and OS-L2-PE are overall the smallest.

From the volatility plots in Figure 4, we see that the structure of the volatility does not change significantly by building dynamic combinations. Both models detect periods of large movements in the log-returns, the main difference being that the AR(1)-DC-GARCH(1,1) model does not reach so large values as the AR(1)-GARCH(1,1) model. An interesting aspect of our model is its capability to sometimes descend more rapidly, or being less persistent, and also to reach low values (see for instance the outsample volatility around sample point 900), in contrast to the AR(1)-GARCH(1,1) model. In Figure 5 we

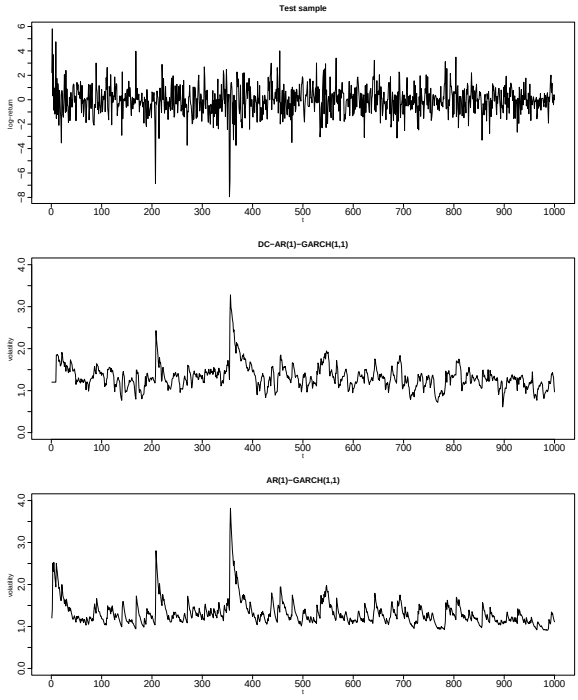


Figure 4: BMW data. From top to bottom: log-returns in test sample, outsample volatility for AR(1)-DC-GARCH(1,1), outsample volatility for AR(1)-GARCH(1,1). Both of the latter with normal innovations.

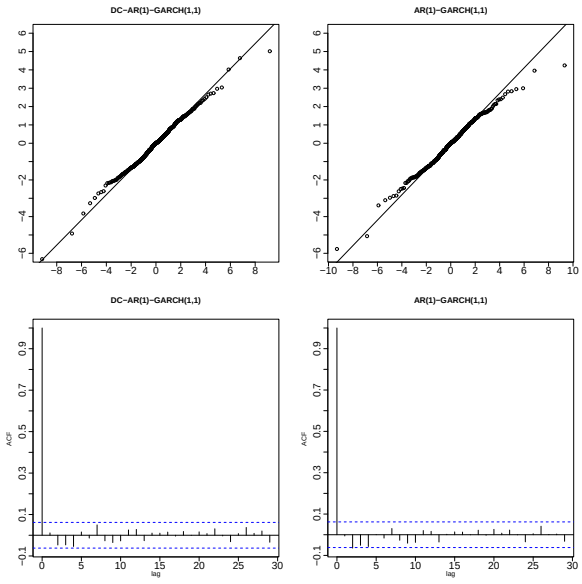


Figure 5: BMW data. Top: normal QQ-plots of outsample residuals. Bottom: autocorrelation function of absolute values of outsample residuals. Left: AR(1)-DC-GARCH(1,1) with normal innovations. Right: AR(1)-GARCH(1,1) with normal innovations.

display graphical diagnostics for the residuals of the selected AR(1)-DC-GARCH(1,1) model and for the AR(1)-GARCH(1,1) model, both for standard normal innovations. The DC-model exhibits more normally distributed residuals while the serial correlations of absolute residuals are low for both models.

More empirical results are given Ferrari (2002). In particular, for daily log-returns from the "Deutscher Aktien-Index" (DAX), our data-driven approach selected the standard AR(1)-GARCH(1,1) over the new AR(1)-DC-GARCH model. This selection turned out to be reasonable because the best AR(1)-DC-GARCH model which minimizes out-sample performance was only slightly better than the AR(1)-GARCH(1,1) fit.

# 6 Conclusions

We proposed a new method for stationary nonlinear time series analysis which combines local models, either parametric or nonparametric, by using mixture probabilities from variable length Markov chains. The approach is very general and flexible: it can be used for modelling conditional means, conditional variances or conditional densities, and the methodology can be applied to almost any kind of local model. It can thus be viewed also as a method which has the potential to improve a given model class, for example ARMA, nonparametric additive autoregressive or GARCH, via dynamic combination.

We present a fully data-driven approach for estimation, discretization (localization) and model selection. Various empirical results illuminate the competitiveness or superior predictive performance of the new method for conditional mean and conditional variance estimation given the previous lagged values. For the former, our Gaussian DC-AR amounts to estimating a VLMC and a convex parameter optimization having a unique solution which can be computed efficiently, in contrast to say projection pursuit as a sophisticated nonparametric alternative. Comparisons are made on synthetic data and with nonparametric techniques such as additive modelling or projection pursuit. For conditional variance models, we consider volatility estimation for a real financial time series and compare it with the GARCH(1,1) model predictions. Finally, a consistency result for dynamic combination of autoregressive models represents some asymptotic aspects.

A generic part of our method is the fitting of variable length Markov chains for which our software is publicly available in the statistical computing language *R* (http://www.r-project.org).

# References


[1] Audrino, F. and Bühlmann, P. (2001). Tree-Structured Generalized Autoregressive Conditional Heteroscedastic Models. Journal of the Royal Statistical Society (Series B) **63**, 727–744.

[2] Bollerslev, T. (1986). Generalized Autoregressive Conditional Heteroscedasticity. Journal of Econometrics **31**, 307–327.

[3] Bühlmann, P. (1999). Dynamic Adaptive Partitioning for Nonlinear Time Series. Biometrika **86**, 555–571.

[4] Bühlmann, P. and Wyner, A. (1999). Variable Length Markov Chains. The Annals of Statistics **27**, 480–513.

[5] Ferrari, F. (2002). Variable length Markov chains and dynamic combination of models. PhD Dissertation No. 14503, ETH Zürich.

[6] Ferrari, F. and Wyner A. (2002). Estimation of General Stationary Processes by Variable Length Markov Chains. To appear in Scandinavian Journal of Statistics.

[7] Friedman, J. and Stuetzle, W. (1981). Projection Pursuit Regression. Journal of the American Statistical Association **76**, 817–823.

[8] Hastie, T. and Tibshirani, R. (1990). *Generalized Additive Models.* Chapman & Hall, London.

[9] Jordan, M.I. and Jacobs, R.A. (1994). Hierarchical mixtures of experts and the EM algorithm. Neural Computation **6**, 181–214.

[10] Mächler, M. and Bühlmann, P. (2002). Variable Length Markov Chains: Methodology, Computing and Software. Preprint. ETH Zürich.

[11] Nocedal, J. and Wright, S.J. (1999). *Numerical Optimization.* Springer, New York.

[12] Rissanen, J. (1983). A Universal Data Compression System. IEEE Transactions in Information Theory **29**, 656–664.

[13] Tjøstheim, D. (1994). Non-linear Time Series: A Selective Review. Scandinavian Journal of Statistics **21**, 97–130.

[14] Tong, H. (1990). *Non-linear Time Series: a Dynamical System Approach.* Oxford: Clarendon Press.

[15] Tong, H. and Lim, K.S. (1980). Threshold autoregression,limit cycles and cyclical data. Journal Royal Statistical Society (Series B) **42**, 245–292.

[16] van der Vaart, A.W. (1998). *Asymptotic Statistic.* Cambridge University Press.

[17] Yu, B. (1994). Rates of convergence for empirical processes of stationary mixing sequences. Annals of Probability **22**, 94–116.

## Appendix A: The context algorithm

The tree structured context algorithm uses the notion of terminal node context trees

$$\tau^T = \tau_c^T = \{w; w \in \tau_c \text{ and } wu \notin \tau_c \text{ for all } u \in \mathcal{X}\}.$$

In Example 2, $\tau^T = \{0, 1, 2, 33\}$ is the set of terminal nodes in the tree in Figure 1, whereas $\tau = \tau^T \cup \{3\}$. In Example 1, $\tau^T$ and $\tau$ coincide. The information of $\tau^T$ is equivalent to the information in $\tau$: the terminal node tree thus yields a more compact representation. Denote by

$$N(w) = \sum_{t=1}^{n-\text{card}(w)+1} 1_{[X_t^{t+\text{card}(w)-1}=w]}, \; w \in \cup_{m=1}^{\infty} \mathcal{X}^m,$$

the number of occurrences of the string $w$ in the sequence $X_1^n$ and let $N_-(w)$ be the same but summing from $t = 1, 2, \ldots, n - \text{card}(w)$. Moreover, let

$$\hat{P}(w) = N(w)/n,$$
$$\hat{P}(x|w) = \frac{N(xw)}{N_-(w)}, \quad x \in \mathcal{X},\ w \in \cup_{m=1}^{\infty}\mathcal{X}^m \tag{A.1}$$

so that $\sum_x \hat{P}(x|w) = 1$.

The algorithm below constructs the estimated context tree $\hat{\tau}$ as the biggest context tree (with respect to the order '$\preceq$' defined in Step 1 below) such that

$$\Delta_{wu} = \sum_{x\in\mathcal{X}} \hat{P}(x|wu)\log(\frac{\hat{P}(x|wu)}{\hat{P}(x|w)})N(wu) \geq K \text{ for all } wu \in \hat{\tau}^T\ (u \in \mathcal{X})$$

where $K$ is the cutoff tuning parameter.

**Step 1** Given $\mathcal{X}$-valued data $X_1, \ldots, X_n$, fit a maximal context tree, i.e., search for the context function $c_{max}(\cdot)$ with terminal node context tree representation $\tau_{max}^T$, where $\tau_{max}^T$ is the biggest tree such that every element (terminal node) in $\tau_{max}^T$ has been observed at least twice in the data. This can be formalized as follows:
$\tau_{max}^T$ is such that $w \in \tau_{max}^T$ implies $N(w) \geq 2$, and such that for every $\tau^T$, where $w \in \tau^T$ implies $N(w) \geq 2$, it holds that $\tau^T \preceq \tau_{max}^T$.
Here, $\tau_1 \preceq \tau_2$ means: $w \in \tau_1 \Rightarrow wu \in \tau_2$ for some $u \in \cup_{m=0}^{\infty}\mathcal{X}^m$ ($\mathcal{X}^0 = \emptyset$).
Set $\tau_{(0)}^T = \tau_{max}^T$.

**Step 2** Examine every element (terminal node) of $\tau_{(0)}^T$ as follows (the order of examining is irrelevant). Let $c(\cdot)$ be the corresponding context function of $\tau_{(0)}^T$ and let

$$wu = x_{-\ell+1}^0 = c(x_{-\infty}^0),\ u = x_{-\ell+1},\ w = x_{-\ell+2}^0,$$

where $wu$ is an element (terminal node) of $\tau_{(0)}^T$, which we compare with its pruned version $w = x_{-\ell+2}^0$ (if $\ell = 1$, the pruned version is the empty branch $\emptyset$, i.e., the root node).
Prune $wu = x_{-\ell+1}^0$ to $w = x_{-\ell+2}^0$ if

$$\Delta_{wu} = \sum_{x\in\mathcal{X}} \hat{P}(x|wu)\log(\frac{\hat{P}(x|wu)}{\hat{P}(x|w)})N(wu) < K,$$

with $K = K_n \sim C\log(n),\ C > 2\text{card}(\mathcal{X}) + 3$ and $\hat{P}(\cdot|\cdot)$ as defined in (A.1). Decision about pruning for every terminal node in $\tau_{(0)}^T$ yields a (possibly) smaller tree $\tau_{(1)} \preceq \tau_{(0)}^T$. Construct the terminal node context tree $\tau_{(1)}^T$.

**Step 3** Repeat Step 2 with $\tau_{(i)}, \tau_{(i)}^T$ instead of $\tau_{(i-1)}, \tau_{(i-1)}^T$ ($i = 1, 2, \ldots$) until no more pruning is possible. Denote this maximal pruned context tree (not necessarily of terminal node type) by $\hat{\tau} = \tau_{\hat{c}}$ and its corresponding context function by $\hat{c}(\cdot)$.

**Step 4** If interested in probability distributions, estimate the transition probabilities $P(x_1|c(x_{-\infty}^0))$ by $\hat{P}(x_1|\hat{c}(x_{-\infty}^0))$, where $\hat{P}(\cdot|\cdot)$ is defined as in (A.1).

More details and motivation can be found in Bühlmann and Wyner (1999).

# Appendix B: Proofs

*Form of the design matrix A for $\hat{\theta}_{LS}$ in (3.11):*

$$\begin{pmatrix}
1 & \hat{P}_{s+1,0} & \dots & \hat{P}_{s+1,N-2} & Y_s & Y_s\hat{P}_{s+1,0} & \dots \\
1 & \hat{P}_{s+2,0} & \dots & \hat{P}_{s+2,N-2} & Y_{s+1} & Y_{s+1}\hat{P}_{s+2,0} & \dots \\
\vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\
\vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\
1 & \hat{P}_{n,0} & \dots & \hat{P}_{n,N-2} & Y_{n-1} & Y_{n-1}\hat{P}_{n,0} & \dots
\end{pmatrix}$$

$$\begin{pmatrix}
\dots & Y_s\hat{P}_{s+1,N-2} & \dots & Y_1 & Y_1\hat{P}_{s+1,0} & \dots & Y_1\hat{P}_{s+1,N-2} \\
\dots & Y_{s+1}\hat{P}_{s+2,N-2} & \dots & Y_2 & Y_2\hat{P}_{s+2,0} & \dots & Y_2\hat{P}_{s+2,N-2} \\
\vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\
\vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\
\dots & Y_n\hat{P}_{n,N-2} & \dots & Y_{n-s} & Y_{n-s}\hat{P}_{n,0} & \dots & Y_{n-s}\hat{P}_{n-s,N-2}
\end{pmatrix}$$

*Proof of Theorem 3.1:*
In the sequel, we often write for a probability measure $P$ on $\mathcal{X}^{\mathbb{Z}}$, $P(x) = \mathbb{P}_P[X_1^q = x]$ $(x \in \mathcal{X}^q)$ [abbreviating $P^{(q)}(x)$] and $P(x|w) = P(xw)/P(w)$ $(x, w \in \cup_{j=1}^{\infty}\mathcal{X}^j)$. The following assumptions are used for proving Theorem 3.1.

(B1) The data-generating process $\{Y_t : t \in \mathbb{Z}\}$ is strictly stationary and geometrically $\beta$-mixing with $\beta(k) \le C_\beta \rho^k$ for some constants $C_\beta > 0$ and $0 < \rho < 1$.

(B2) $\mathbb{E}|Y_t|^{2+\kappa} < \infty$ for some $\kappa > 0$.

(B3) For the discretized process $\{X_t = q(Y_t); t \in \mathbb{Z}\}$, consider the sequence of truncated context functions

$$c_n(x_{-\infty}^0) = \begin{cases} c(x_{-\infty}^0) & \text{if } w = c(x_{-\infty}^0) \text{ has length } \mathrm{card}(w) \le d_n \\ x_{-d_n+1}^0 & \text{if } \mathrm{card}(c(x_{-\infty}^0)) > d_n \end{cases}, \tag{A.2}$$

for an increasing sequence $\{d_n : n \in \mathbb{N}\}$. The corresponding context and terminal node context trees (see section 2.2) are denoted by $\tau_n$ and $\tau_n^T$, respectively. We then assume:

(a) For all n sufficiently large

$$d_n \le n^\delta, \text{ for some } \delta \in (0, \sigma),$$

where $\sigma \in (0,1)$ is specified in assumption (B3(b)).

(b) For some $\theta > 0$, some $\sigma \in (0,1)$ and some $\gamma \in (0, (1-\sigma)/2)$, for all n sufficiently large,

$$\Gamma_n = \min_{w \in \tau_n^T} P(w) \ge \frac{1}{n^\gamma},$$

$$\Upsilon_n = \min_{wu \in \tau_n^T, u \in \mathcal{X}} \sum_{x \in \mathcal{X}} |P(x|wu) - P(x|w)|, \quad \Upsilon_n^2 \ge \frac{\log(n)^{1+\theta}}{\left(n\Gamma_n^{(1-\sigma)/2}\right)^{1-\sigma}}.$$

(c) For the minimal transition probabilities, for all n sufficiently large,

$$P_{min}(n) = \min_{x \in \mathcal{X}, w \in \tilde{\tau}_n} P(x|w) \ge \frac{1}{n}.$$

(C) The context algorithm is used with the estimated context function $\hat{c}_n$ but truncated at $d_n$ as in (A.2). Moreover, the cutoff tuning parameter satisfies $K = K_n \sim C\log(n)$, $C > 2\mathrm{card}(\mathcal{X}) + 3$.

Assumptions (B3(a))-(B3(c)) are all probabilistic conditions about the sparseness and the growth rate of the truncated context tree $\tau_n$ and the corresponding set of terminal nodes $\tau_n^T$.

They may be hard to check, but Ferrari and Wyner (2000) give an example where they hold. For general stationary processes with finite memory (being therefore VLMC's), it suffices to assume

$$\min_{x\in\mathcal{X},w\in\tau} P(x|w) > 0$$

which implies assumptions (B3(a)) - (B3(c)).

*Proof of Theorem 3.1.* The least squares criterion is

$$\hat{L}_n(\theta) = \frac{1}{n-s}\sum_{t=s+1}^{n}(Y_t - \hat{\mu}_t(\theta))^2.$$

Consider also the auxiliary functions

$$L_n(\theta) = \frac{1}{n-s}\sum_{t=s+1}^{n}(Y_t - \mu_t(\theta))^2$$

which replaces the estimated $\hat{P}_{t,x}$ by the true underlying $P_{t,x}$, and

$$L(\theta) = \mathbb{E}[(Y_t - \mu_t(\theta))^2].$$

Also note that the best projected parameter $\theta_*$ is unique due to convexity of $L(\theta)$ (see Nocedal and Wright (1999)). Since $\hat{\theta}_{LS,n}$ is the minimizer of $\hat{L}_n(\theta)$, it suffices to show

$$\sup_{\theta\in\Theta}|\hat{L}_n(\theta) - L(\theta)| = o_P(1). \tag{A.3}$$

See for example van der Vaart (1998, Th. 5.7). By the triangle inequality we have

$$\begin{aligned} & \sup_{\theta\in\Theta}|\hat{L}_n(\theta) - L(\theta)| \\ \leq\ & \sup_{\theta\in\Theta}|\hat{L}_n(\theta) - L_n(\theta)| + \sup_{\theta\in\Theta}|L_n(\theta) - L(\theta)| = I + II. \end{aligned} \tag{A.4}$$

A first order Taylor expansion yields for the first term $I$,

$$I = \sup_{\theta}\left|-2\frac{1}{n-s}\sum_{t=1}^{n-s}(Y_t - \mu_t(\theta;\tilde{P}_{\cdot,x}))\sum_{x=0}^{N-1}(\theta_{x,0} + \sum_{j=1}^{p}\theta_{x,j}Y_{t-j})(\hat{P}_{t,x} - P_{t,x})\right|,$$

$$\mu_t(\theta;\tilde{P}_{\cdot,x}) = (\theta_{x,0} + \sum_{j=1}^{p}\theta_{x,j}Y_{t-j})\tilde{P}_{t,x}.$$

where $|\tilde{P}_{t,x} - P_{t,x}| < |\hat{P}_{t,x} - P_{t,x}|$. This expression can be bounded as

$$I \leq 2\sup_{t,x}|\hat{P}_{t,x} - P_{t,x}|\frac{1}{n-s}\sum_{t=1}^{n-s}\left(|Y_t| + \sup_{\theta}|\mu_t(\theta;\tilde{P}_{\cdot,x})|\right)\cdot\sup_{\theta}\sum_{x=0}^{N-1}\left(|\theta_{x,0}| + \sum_{j=1}^{p}|\theta_{x,j}||Y_{t-j}|\right).$$

It is shown in Ferrari and Wyner (2000) that $\sup_{t,x}|\hat{P}_{t,x} - P_{t,x}| = o_P(1)$. Thus, due to the fact that $\Theta$ is compact and the moment assumption in (B2) which ensures a law of large numbers, it follows that $I = o_P(1)$.

In order that the second term in (A.4) satisfies $II = o_P(1)$, it suffices to show that the following conditions for $m_\theta(Y_t,\ldots,Y_{t-s}) = (Y_t - \mu_t(\theta))^2$ hold (see for example van der Vaart (1998, Th. 19.4 and Ex. 19.8) and Yu (1994)):

(D1) The function $\theta \to m_\theta(Y_t,\ldots,Y_{t-s})$ is continuous for every $(Y_t,\ldots,Y_{t-s}) \in \mathbb{R}^{s+1}$.

(D2) For the family, there $\{m_\theta : \theta\in\Theta\}$, there exists an integrable envelope function $b(\cdot)$ such that $\sup_{\theta\in\Theta}|m_\theta(Y_t,\ldots,Y_{t-s})| \leq b(Y_y,\ldots,Y_{t-s})$.

The condition (D1) holds since $\mu(t,\theta)$ is linear in $\theta$. For the condition (D2), define the envelope function

$$b(Y_t,\ldots,Y_{t-s}) = \sup_{\theta\in\Theta}(Y_t - \mu_t(\theta))^2.$$

Since $\Theta$ is compact and due to the moment assumption (B2), the envelope function $b(\cdot)$ is integrable. □