Tree Structured GARCH Models

Francesco Audrino and Peter Bühlmann*
ETH Zürich, Switzerland

March 2000

Abstract

We propose a new GARCH model with tree-structured multiple thresholds for volatility estimation in financial time series. The approach relies on the idea of a binary tree where every terminal node parameterizes a (local) GARCH model for a partition cell of the predictor space. Fitting of such trees is constructed within the likelihood framework for non-Gaussian observations: it is very different from the well-known CART procedure for regression based on residual sum of squares. Our strategy includes the classical GARCH model and allows in a systematic and flexible way to increase model-complexity. We conclude with simulations and real data analysis that the new method has better predictive potential compared to other approaches.

Keywords: Conditional variance; Financial time series; GARCH model; Maximum likelihood; Threshold model; Tree model.

1 Introduction

We propose a new method for estimating volatility in stationary financial time series; our real data examples are daily log-returns $X_t = \log(P_t/P_{t-1})$, where P_t denotes the price of an asset at day t. The modeling technique is parametric and potentially high-dimensional: it adds thresholds, and thus partitions a predictor space, in the fashion of a binary tree. Although the binary tree reminds a bit to CART (Breiman et al., 1984), our approach is very different; we explain this below. And our modeling scheme is markedly different from autoregressive threshold models (SETAR) for conditional expectations, cf. Tong (1990): because we consider the conditional variance and we also allow for non-Markovian models.

As a starting point, consider a nonparametric GARCH(1,1) model,

$$\begin{aligned} &X_t = \sigma_t Z_t \ (t \in \mathbb{Z}), \\ &\sigma_t^2 = f(X_{t-1}, \sigma_{t-1}^2),\ f: \mathbb{R} \times \mathbb{R}^+ \to \mathbb{R}^+, \end{aligned} \tag{1.1}$$

where $(Z_t)_{t\in\mathbb{Z}}$ is a sequence of i.i.d. innovation variables with $\mathbb{E}[Z_t] = 0$, $\mathrm{Var}(Z_t) = 1$ and Z_t independent from $\{X_s; s < t\}$. The so-called volatility σ_t is then related as

$$\sigma_t^2 = \mathrm{Var}(X_t|\mathcal{F}_{t-1}) = f(X_{t-1}, \sigma_{t-1}^2),$$

where $\mathcal{F}_{t-1}$ denotes the σ-algebra (the information) of the variables $\{X_s; s \le t-1\}$. The restriction to model the squared volatility σ_t^2 as a function of the previous values X_{t-1} and σ_{t-1}^2 alone

*Corresponding author. Address: Seminar für Statistik, LEO D12, ETH Zentrum, CH-8092 Zürich, Switzerland. E-mail: buhlmann@stat.math.ethz.ch.

is natural in finance. Note that it still generates a dependence of σ_t^2 from *all previous* observations $\{X_s; s < t\}$ due to the recursive definition with σ_{t-1}^2: this is the important mathematical difference between ARCH and GARCH models. Also, the assumption that $\mathbb{E}[X_t|\mathcal{F}_{t-1}] \equiv 0$ is a reasonable approximation for many financial time series: the substantial modeling effort goes into the dominant volatility (for real data, we usually subtract first a linear AR(1) estimate for the conditional mean). The simplest but often used example for (1.1) is the classical GARCH(1,1) model (Bollerslev, 1986),

$$f(x, \sigma^2) = \alpha_0 + \alpha_1 x^2 + \beta\sigma^2, \ \alpha_0, \alpha_1, \beta > 0. \tag{1.2}$$

The unknown function $f(\cdot,\cdot)$ in (1.1) may be nonlinear and even not smooth; for example, with an asymmetry in financial trading around positive and negative return values leading to a discontinuous behavior for $f(x, y)$ around $x = 0$. Estimation of $f(\cdot,\cdot)$ in generality is very difficult due to the non-observable volatility in the second argument. An iterative nonparametric estimation procedure has been proposed in Bühlmann and McNeil (1999). It has the usual advantage about flexibility for $f(\cdot,\cdot)$ (although quite a few theoretical issues are not rigorously settled yet); the disadvantages are mainly: lack of performance at edges which include the high values of volatility being of particular interest in practice; complications when dealing with non-Gaussian observations; sensitivity to choice of smoothing parameter. Our approach here is more in the spirit of a *sieve approximation* with parametric models for the nonparametric function $f(\cdot,\cdot)$. The approximation has the following principles:

(1) It includes the classical GARCH(1,1) model as a simple special case,

(2) It uses a binary tree type selection strategy to determine splits (thresholds) for building up an approximating multiple threshold GARCH model. The binary tree construction, where every terminal node represents a (local) three-dimensional GARCH model, is based on the likelihood in model (1.1).

Issue (1) has an important link to practice: there is a quite strong believe that the classical GARCH(1,1) model is appropriate, despite its simplicity with three parameters. Our tree structured nested modeling strategy allows to verify such a hypothesis by using known selection techniques for nested models: as we will see, there is potential to improve upon the classical GARCH(1,1) for real data and we will quantify such gains in terms of prediction accuracy for volatility (rather than testing about structural properties of $f(\cdot,\cdot)$).

The likelihood driven tree method mentioned in issue (2) marks an essential difference to CART (Breiman et al., 1984): underlying an approximate normality assumption for observations, CART uses residual sum of squares. For financial data, the normality assumption for observations and the corresponding techniques are not appropriate at all and can result in very poor performance. Our approach resembles more the general tree fitting with the deviance criterion used by Clark and Pregibon (1993). Another difference to CART (or more general versions driven by deviance) is that our tree structured scheme employs a three-dimensional (local) GARCH model for every terminal node in the binary tree; CART uses only one location parameter per node. Finally, our tree GARCH is for modeling the function $f(\cdot,\cdot)$ in (1.1) and hence for the *infinite past* in terms of the observations; whereas CART (or versions thereof) in autoregressive modeling deals with a p-dimensional predictor space ($p < \infty$) from finitely many lagged observations.

Extending the GARCH(1,1) model in (1.2) in the direction of adding potentially high parametric complexity hasn't been considered yet. Other versions of GARCH(1,1) with three or four parameters and a GARCH model with one or two thresholds at *fixed* locations (Rabemananjara and Zakoian, 1993) exist already. But there seems to be no systematic flexible route how to build up a class of models from the classical GARCH(1,1) in (1.2) to (a potentially high dimensional approximation of) the general model in (1.1). The paper here deals mainly with this latter task: we describe how this can be done and we demonstrate its power and use on simulated and real data.

2 Tree structured GARCH estimation

We describe here our methodology for approximating $f(\cdot,\cdot)$ in (1.1) by a piecewise linear function (a threshold model). The novel part thereby is the estimation of thresholds in $\mathbb{R}\times\mathbb{R}^+$, where piecewise approximation functions begin and end. Our algorithm below is always based on the likelihood in the working model

$$X_t = \phi X_{t-1} + \sigma_t(\theta) Z_t \ (t \in \mathbb{Z}), \tag{2.1}$$

with $\sigma_t(\theta)Z_t$ as in model (1.1), but the functional form $f(\cdot,\cdot)$ now parameterized by a threshold function $f_\theta(\cdot,\cdot)$ which changes in the fitting procedure. We also add here a linear autoregressive term for estimating a conditional mean (being of minor importance for many financial time series). The negative log-likelihood is then

$$\begin{aligned} -\ell(\phi,\theta;X_2^n) &= -\sum_{t=2}^{n} \log\left(\sigma_t^{-1}(\theta) f_Z(\frac{X_t - \phi X_{t-1}}{\sigma_t(\theta)})\right), \\ \sigma_t^2(\theta) &= f_\theta(X_{t-1}, \sigma_{t-1}^2(\theta)), \end{aligned} \tag{2.2}$$

where $f_Z(\cdot)$ denotes the density of the innovation Z_t. The log-likelihood is always considered conditional on X_1 and some reasonable starting value $\sigma_1(\theta)$, e.g. $\sigma_1^2(\theta) = \mathrm{Var}(X_1)$.

The function $f_\theta(\cdot,\cdot)$ is parameterized as binary tree structured GARCH(1,1). It involves a partition

$$\mathcal{P} = \{\mathcal{R}_1, \ldots \mathcal{R}_k\},\ \cup_{j=1}^k \mathcal{R}_j = \mathbb{R}\times\mathbb{R}^+,\ \mathcal{R}_i \cap \mathcal{R}_j = \emptyset\ (i \neq j)$$

for the predictor space. Then, for every partition cell $\mathcal{R}_j$, we employ a GARCH(1,1) model: the parametric form of the function then depends on $\mathcal{P}$,

$$f_\theta(x,\sigma^2) = f_\theta^{\mathcal{P}}(x,\sigma^2) = \sum_{j=1}^{k} \left(\alpha_{0,j} + \alpha_{1,j}x^2 + \beta_j \sigma_{t-1}^2\right) I_{[(x,\sigma^2)\in\mathcal{R}_j]}, \tag{2.3}$$

where θ denotes the parameter set $\{\alpha_{0,j}, \alpha_{1,j}, \beta_j;\ j = 1, \ldots, k\}$. For $k = 1$, we have the classical GARCH(1,1) model from (1.2). The partition $\mathcal{P} = \{\mathcal{R}_1, \ldots, \mathcal{R}_k\}$ is constructed as a binary tree (every terminal node representing a partition cell $\mathcal{R}_j$) as follows. A first threshold $d_1 \in \mathbb{R}$ or $\mathbb{R}^+$ together with a component index $\iota_1 \in \{1,2\}$ partitions

$$\mathbb{R}\times\mathbb{R}^+ = \mathcal{R}_{left} \cup \mathcal{R}_{right},$$

where $\mathcal{R}_{left} = \{(x,\sigma^2) \in \mathbb{R}\times\mathbb{R}^+;\ (x,\sigma^2)_{\iota_1} \leq d_1\}$ and $\mathcal{R}_{right}$ analogously but with the relation '$>$' instead. Then, one of the partition cells $\mathcal{R}_{left}$ or $\mathcal{R}_{right}$ is again partitioned with a threshold d_2 and a component index ι_2 in the same fashion. We iterate this procedure: for the mth iteration step, we specify a pair (d_m, ι_m) and an existing partition cell for further refining by splitting it into two cells. The refinement of an existing partition is here always constructed according to the following general rule:

$$\begin{aligned}&\mathcal{P}^{(old)} = \cup_j \mathcal{R}_j \text{ an existing partition } \rightarrow \text{ pick an element } \mathcal{R}_{j^*} \in \mathcal{P}^{(old)}\\ &\rightarrow \text{ split } \mathcal{R}_{j^*} = \mathcal{R}_{j^*,left} \cup \mathcal{R}_{j^*,right} \text{ described by } (d,\iota) \in \mathbb{R} \times \{1,2\}\\ &\rightarrow \mathcal{P}^{(new)} = \cup_{j\neq j^*} \mathcal{R}_j \cup (\mathcal{R}_{j^*,left} \cup \mathcal{R}_{j^*,right}),\end{aligned} \tag{2.4}$$

with (d,ι) describing a threshold and component index, $\mathcal{R}_{j^*,left} = \{(x,\sigma^2) \in \mathcal{R}_{j^*} \subset \mathbb{R} \times \mathbb{R}^+; (x,\sigma^2)_\iota \leq d\}$ and analogously for $\mathcal{R}_{j^*,right}$ with the relation '>'. The procedure then produces a partition $\mathcal{P} = \{\mathcal{R}_1, \ldots, \mathcal{R}_k\}$ which can be represented as a binary tree: every terminal node in the tree represents a partition cell $\mathcal{R}_j$. This is conceptually as in CART (Breiman et al., 1984). But the data-driven construction of a tree or partition, including the splitting operations in (2.4), and estimation of parameters are very different.

2.1 Forward entering of thresholds: growing the binary tree

The available univariate data is $X_1, \ldots, X_n$. The algorithm below constructs a binary tree, corresponding to a partition of $\mathbb{R} \times \mathbb{R}^+$, by optimizing reduction of a reduced negative log-likelihood. Thereby, binary trees are equipped with 3 parameters per terminal node for different GARCH(1,1) models for different partition cell.

Step 1. Compute the negative log-likelihood from (2.2) without partitioning (i.e. in the partition $\mathcal{P}^{(0)}_{opt} = \mathbb{R} \times \mathbb{R}^+$) with

$$f^{\mathcal{P}^{(0)}_{opt}}_{\theta^{(0)}}(x,\sigma^2) = \alpha_0 + \alpha_1 x^2 + \beta\sigma^2,\ \theta^{(0)} = (\alpha_0, \alpha_1, \beta),$$

and derive from it the maximum likelihood estimates $\hat{\phi}^{(0)}, \hat{\theta}^{(0)}$ using a quasi-Newton method, cf. Nocedal and Wright (1999). Set $m = 0$.

Step 2. Increment m by one. Search for the best refined partition $\mathcal{P}^{(m)}_{opt}$ by binary splitting of a cell from $\mathcal{P}^{(m-1)}$ as described in (2.4). The details are as follows:
(I) Given $\mathcal{P}^{(m-1)}_{opt} = \{\mathcal{R}_1, \ldots, \mathcal{R}_m\}$, consider a new partition $\mathcal{P}^{(m)}$, where one partition cell $\mathcal{R}_{j^*} \in \mathcal{P}^{(m-1)}$ is split into $\mathcal{R}_{j^*} = \mathcal{R}_{j^*,left} \cup \mathcal{R}_{j^*,right}$. The function associated with $\mathcal{P}^{(m)}$ is

$$\begin{aligned} f^{\mathcal{P}^{(m)}}_{(\theta^{(m-1)\setminus *},\theta^*)}(x,\sigma^2) &= \sum_{j\neq j^*} (\alpha_{0,j} + \alpha_{1,j}x^2 + \beta_j\sigma^2) I_{[(x,\sigma^2)\in\mathcal{R}_j]}\\ &+ \sum_{i\in\{j^*_{left}, j^*_{right}\}} (\alpha_{0,i} + \alpha_{1,i}x^2 + \beta_i\sigma^2) I_{[(x,\sigma^2)\in\mathcal{R}_i]}, \end{aligned} \tag{2.5}$$

where

$$\begin{aligned}&\theta^{(m-1)\setminus *} = \{\alpha_{0,j}, \alpha_{1,j}, \beta_j;\ j = 1, \ldots, m,\ j \neq j^*\}\\ &\theta^* = \{\alpha_{0,i}, \alpha_{1,i}, \beta_i;\ i \in \{j^*_{left}, j^*_{right}\}\}.\end{aligned}$$

(II) Compute the minimal negative reduced log-likelihood in the refined partition $\mathcal{P}^{(m)}$,

$$\min_{\theta^*} \left(-\ell^{\mathcal{P}^{(m)}}(\hat{\phi}^{(0)}, (\hat{\theta}^{(m-1)\setminus *}, \theta^*); X_2^n) \right), \tag{2.6}$$

by numerical minimization over θ^* using a quasi-Newton method. Here, $-\ell^{\mathcal{P}^{(m)}}$ is as in (2.2) with the function $f^{\mathcal{P}^{(m)}}(\cdot,\cdot)$ from (2.5). For this numerical minimization over θ^*, use as starting

values the components of $\hat{\theta}^{(m-1)}$ for cell $\mathcal{R}_{j^*}$ for both $(\theta_1^*, \theta_2^*, \theta_3^*)$ and $(\theta_4^*, \theta_5^*, \theta_6^*)$ (the parameters for the refined partition cells $\mathcal{R}_{j^*,left}, \mathcal{R}_{j^*,right}$).
(III) Optimize (2.6) by varying $\mathcal{P}^{(m)}$ in (I) and recomputing (II). Denote the optimal refined partition by $\mathcal{P}_{opt}^{(m)}$.

Step 3. Compute the maximum likelihood in partition $\mathcal{P}_{opt}^{(m)}$. Minimize with a quasi-Newton method the negative log-likelihood in (2.2) with $f^{\mathcal{P}_{opt}^{(m)}}(\cdot,\cdot)$ from (2.5) to obtain $(\hat{\phi}^{(m)}, \hat{\theta}^{(m)})$. Thereby, use the starting values $\hat{\phi}^{(m-1)}$, $\hat{\theta}^{(m-1)\setminus *}$ and $\hat{\theta}^*$ being the minimizer for (2.6) which is computed in Step 2.

Step 4. Repeat Steps 2 and 3 until $m = M$. This yields a partition $\mathcal{P}_{opt}^{(M)}$ corresponding to a large binary tree equipped with parameter estimates $(\hat{\phi}^{(M)}, \hat{\theta}^{(M)})$.

Remark 2.1. The value M, corresponding to $M+1$ partition cells or terminal nodes in the binary tree, is pre-specified in advance such that the binary tree is sufficiently large. With financial return data, choosing M around 6 is often sufficient.

Remark 2.2. The search for the splitting value in Step 3 is done on a grid: we propose grid-points being empirical α-quantiles of the data with $\alpha = i/\text{mesh},\ i = 1, \ldots, \text{mesh} - 1$. We typically choose mesh $= 8$ or 16.

Remark 2.3. The reduced log-likelihood in (2.6) yields a substantial computational short-cut compared to a full likelihood approach. For every given partition $\mathcal{P}^{(m)}$, the numerical nonlinear minimization in (2.6) involves only the 6-dimensional parameter θ^*. Since our algorithm searches over many candidate partitions $\mathcal{P}^{(m)}$ in every iteration step m, a relatively fast nonlinear minimization is important. Finding the best split is thus determined by maximal reduction of the negative reduced log-likelihood.

Remark 2.4. The parameter estimates in Step 3 are computed from the full likelihood. For $(\hat{\phi}^{(m)}, \hat{\theta}^{(m)})$ we take advantage of the fact that the starting values specified in Step 3 are very reasonable for obtaining a reliable and fast maximum likelihood estimate in a possibly high-dimensional parameter space.

2.2 Pruning the tree

The binary tree, or the partition $\mathcal{P}_{opt}^{(M)}$, constructed in section 2.1 is too large, or too fine, respectively. We thus prune back: since M around 6 seems large enough for very many financial time series, searching for the best subtree is computationally feasible. Denote by τ the set of all binary subtrees from $\mathcal{P}_{opt}^{(M)}$: its elements are denoted as $\mathcal{P}_i$. Thus, every $\mathcal{P}_i$ corresponds to a partition of $\mathbb{R} \times \mathbb{R}^+$ which can be represented as a binary tree. Note that τ is generally larger than the set $\{\mathcal{P}_{opt}^{(0)}, \mathcal{P}_{opt}^{(1)}, \ldots, \mathcal{P}_{opt}^{(M)}\}$.

For every $\mathcal{P}_i$, we compute the maximum likelihood estimates $\hat{\phi}^{\mathcal{P}_i}, \hat{\theta}^{\mathcal{P}_i}$ with a quasi-Newton method, according to (2.2) with $f^{\mathcal{P}_i}(\cdot,\cdot)$ of the form (2.3). Note that reasonable starting values are again at hand by going backwards from $\hat{\phi}^{(M)}, \hat{\theta}^{(M)}$ in a stage-wise manner. We then consider the penalized negative log-likelihood

$$-2\ell(\hat{\phi}^{\mathcal{P}_i}, \hat{\theta}^{\mathcal{P}_i}; X_2^n) + 2(\dim(\hat{\theta}^{\mathcal{P}_i}) + 1) \tag{2.7}$$

as a measure for predictive potential, namely the AIC statistic. The additional contribution 1 in the penalty term arises whenever the conditional expectation parameter ϕ in (2.1) is estimated. Choose the binary tree, or the partition $\hat{\mathcal{P}}$, minimizing (2.7). The final tree structured GARCH model is thus given by (2.1) with $\hat{\phi}^{\hat{\mathcal{P}}}$, and $f^{\hat{\mathcal{P}}}(\cdot,\cdot)$ as in (2.3) based on partition $\hat{\mathcal{P}}$ and parameter estimates $\hat{\theta}^{\hat{\mathcal{P}}}$.

Remark 2.5. The AIC-statistic in (2.7) can be replaced by any other sensible model selection criterion. We have experimented with two other versions but found that overall performance with AIC is very satisfactory.

3 Numerical results

In this section, we consider the performance of tree structured GARCH models for simulated and real data. We compare some of the results with the GARCH(1,1) model in (1.2) and a nonparametric generalized additive model for log-transformed squared data which is described in the Appendix. We always report here with the use of $M = 5$ in Step 4 from section 2.1 and with grid search as described in Remark 2.2 having mesh = 8 (except in Table 3.2 with mesh = 16 in addition): these specifications already lead to good tree structured model fits, despite their somewhat simple nature. For numerical optimization, we use the quasi-Newton method from the function '*nlmin*' implemented in S-Plus.

3.1 Simulations

The model that we use for the simulations is as in (2.1) with $\phi = 0$ and

$$f(x,\sigma^2) = \begin{cases} \alpha_{0,1} + \alpha_{1,1}x^2 + \beta_1\sigma^2 \ , & \text{if } x \leq d_1, \\ \alpha_{0,2} + \alpha_{1,2}x^2 + \beta_2\sigma^2 \ , & \text{if } x > d_1 \text{ and } \sigma^2 \leq d_2, \\ \alpha_{0,3} + \alpha_{1,3}x^2 + \beta_3\sigma^2 \ , & \text{if } x > d_1 \text{ and } \sigma^2 > d_2, \end{cases} \tag{3.1}$$

with the following parameters and thresholds d_1 and d_2,

$$\alpha_{0,1} = 0.1,\ \alpha_{0,2} = 0.2,\ \alpha_{0,3} = 0.8,\ \alpha_{1,1} = 0.5,\ \alpha_{1,2} = 0.2,\ \alpha_{1,3} = 0,$$
$$\beta_1 = 0,\ \beta_2 = 0.75,\ \beta_3 = 0.5 \ \text{ and } \ d_1 = 0,\ d_2 = 0.5\ .$$

The parameters $\{\alpha_{0,i},\ \alpha_{1,i}$ and $\beta_i\}$ are chosen to mimic time series of real log-returns. Also, the first threshold $d_1 = 0$ splitting the x-coordinate of the function $f(\cdot,\cdot)$ is natural in finance, saying that volatility behaves differently when the first lagged observation (log-return) is positive or negative. The innovation distribution is chosen as standard normal $Z_t \sim \mathcal{N}(0,1)$ or as scaled t_6 so that $\sqrt{6/4}Z_t \sim t_6$ has again variance one. We always take sample size $n = 1000$: for real daily data, this would correspond to about four years which is a reasonable window in which stationarity is expected to hold approximately.

Estimation is here always based using the knowledge that $\mu_t = \mathbb{E}[X_t|\mathcal{F}_{t-1}] \equiv 0$, i.e. $\phi = 0$ in 2.1. Figures 3.1-2 display some results from the tree structured GARCH model, in comparison with the classical GARCH(1,1) in (1.2) and the GAM model described in the Appendix.

FIGURE 3.1 ABOUT HERE.

From Figure 3.1 we see the following for this particular case. The tree structured GARCH model with $\mathcal{N}(0,1)$ innovations overestimates the number of thresholds: the first two thresholds are approximately correct. An improvement is given by the tree structured GARCH with scaled t_ν-distributed innovations: the thresholds and also the estimated degrees of freedom $\hat{\nu} = 5.12$ for the innovations are very satisfactory. The classical GARCH(1,1) with scaled t_ν-distributed innovations yields $\hat{\nu} = 4.37$ and of course, it doesn't exhibit any 'breakpoints' in the volatility surface. There is no surprise that the tree structured GARCH with scaled t_ν-distributed innovations is best, since the true model is of this form. However, the tree GARCH estimate with $\mathcal{N}(0,1)$ misspecified innovations is as a quasi-maximum-likelihood fit still reasonably good.

FIGURE 3.2 ABOUT HERE

Figure 3.2 exploits a desirable important feature: the tree structured GARCH with scaled t_ν-distributed innovations performs very well in regions of high conditional variance. This is not the case with classical GARCH(1,1); and the GAM fit described in the Appendix is very poor, particularly in regions of high volatility. Note the different scales for the various procedures in Figure 3.2.

For quantifying the goodness of fit, we consider various statistics:

$$\mathrm{L}_i = \sum_{t=1}^{n} \mid \sigma_t^2 - \widehat{\sigma}_t^2 \mid^i, \; i = 1, 2,$$

the AIC statistic from (2.7),

$$\text{OS-L}_i = \sum_{t=1}^{n} \mid \sigma_t^2 - \widehat{\sigma}_t^2(Y_1^{t-1}) \mid^i, \; i = 1, 2, \; Y_1^n \text{ a new test set}$$

where in OS-L (out-sample loss), $\widehat{\sigma}_t^2(Y_1^{t-1})$ is using the estimated model from the data X_1^n and evaluates it on new test data Y_1^{t-1} being another independent realization of the data-generating process. Both, the out-sample OS-L- and AIC-statistic are measures for predictive performance; whereas the L-norms are in-sample quantities. We can't calculate the L-norms and the OS-L-statistics for real data examples, but they are important measures for our simulations. Detailed results are reported in Table 3.1 where we refer to data 1, data 2 and data 3 as three independent realizations from the model (2.1) with (3.1) as described above having $\mathcal{N}(0,1)$ innovations; and data 4 denotes a realization but with scaled innovations $\sqrt{6/4}Z_t \sim t_6$.

Table 3.1 ABOUT HERE.

We view OS-L_2 as the most relevant statistic for simulations. It gives more weight to large deviations than the OS-L_1 criterion, which is often appropriate when studying the estimate in regions of high volatility. The tree structured GARCH procedure consistently outperforms the classical GARCH(1,1) and the nonparametric GAM estimate (for the latter, also the out-sample performance OS-L is very bad and we don't report it explicitly). The classical GARCH(1,1) can exhibit huge variation in out-sample accuracy OS-L, for example a poor performance in data 2. This might be due to a very flat negative log-likelihood at the observed data: see Zumbach (1999) who also proposes a remedy. Interestingly, the tree structured GARCH model exhibits here much more stability.

3.2 Two real data examples

We consider two financial instruments with 1000 daily negative log-returns $X_t = -100 \log(P_t/P_{t-1})$ (in percentages): from the German DAX index between January 18, 1994 and November 17, 1997; and from the BMW stock price between September 23, 1992 and July 23, 1996. We consider the the tree structured GARCH model, again in comparison with the GARCH(1,1) from (1.2) (both with an additional model term ϕX_{t-1} from (2.1) for $\mathbb{E}[X_t|\mathcal{F}_{t-1}]$), and with the GAM model described in the Appendix.

FIGURE 3.3 ABOUT HERE.

Figure 3.3 shows the result for the DAX index from a tree GARCH fit with $\mathcal{N}(0,1)$-distributed innovations. Three thresholds are fitted in the volatility surface. Graphical diagnostics for the residuals is satisfactory: with a tendency for heavier tails than standard normal.

For these real-data examples, we measure goodness of fit with the AIC statistic from (2.7) and the in-sample L_2-loss,

$$\text{IS-L}_2 = \sum_{t=1}^{n} \left(\widehat{\sigma}_t^2 - (X_t - \widehat{\mu}_t)^2 \right)^2 ,$$

with $\hat{\mu}_t = \hat{\phi} X_{t-1}$.

TABLE 3.2 ABOUT HERE.

The tree structured model improves upon the classical GARCH(1,1) (both with $\mathcal{N}(0,1)$-distributed innovations) in the two real-data examples: it is more prominent for the index than for the individual stock price. This is consistent with a common belief that classical GARCH(1,1) is better for individual prices than indices.

3.3 Summarizing numerical results

The tree structured GARCH model consistently outperforms the GARCH(1,1) model in (1.2) and the nonparametric GAM model described in the Appendix: this better performance is with respect to many goodness of fit and graphical criteria. More specifically:

(1) the tree structured GARCH is *much better* in the interesting regions where the true volatility is high, see Figure 3.2.

(2) with the tree structured GARCH, the AIC statistic is consistently lower than for classical GARCH(1,1), see Tables 3.1 and 3.2. The gain in terms of out-sample performance OS-L in simulated examples is even more substantial.

(3) the tree structured fitting procedure might be slightly improved by assuming scaled t_ν-distributed innovations Z_t for the likelihood in (2.2); provided that the underlying innovations are heavier tailed, which is weakly evident in Figure 3.3 for real data. See also Table 3.1.

(4) the AIC-statistic is an indicator for ranking out-sample performance with the OS-L-statistic, see Table 3.1; and pruning with AIC in (2.7) works well.

(5) the nonparametric GAM model described in the Appendix is extremely poor in regions of high volatility (this problem doesn't disappear when trying other smoothing parameters), see Figure 3.2.

Issues (1), (2) and (5) indicate a strong advantage of parametric, likelihood based methods over nonparametric least squares smoothing techniques such as the GAM specification used here or multiplicative nonparametric ARCH models in Hafner (1998) or Yang et al. (1999). As pointed out in (3), the likelihood approach can be easily modified to heavier tailed innovations inducing then even more heavy tails for the observations in the model. If performance is judged with a criterion putting emphasis on accurate prediction in high volatility regions, our tree based GARCH model is clearly best among all alternative methods considered here.

3.4 How appropriate is GARCH(1,1) for daily returns?

The GARCH(1,1) model in (1.2) is very popular for analyzing daily log-returns of financial assets: it is often argued that it performs well despite that it has only three parameters describing a very low-dimensional model for sample size in the range of 1000. We quantify here the possible gains by using the flexible tree structured GARCH model. In virtually all examples of daily log-return stock data, the new tree GARCH procedure improves upon the classical GARCH(1,1): the difference in performance has often quantitative magnitude as reported in sections 3.1- 3.3, with a remarkable gain for regions of high volatility. With real data, the first split has always been found in the x-axis around zero: this is compatible with the interpretation that there is an asymmetric behavior depending on the sign of the previous log-return value.

4 Concluding remarks

We have presented a tree structured GARCH model which is more flexible and accurate for prediction of volatility in financial time series than classical GARCH(1,1). The modeling strategy includes the classical GARCH(1,1) as a special case (no thresholds) and allows to increase complexity in a *systematic* way. Also, the new method compares very favorably with a nonparametric technique based on additive models: especially in the interesting regions where the true volatility is moderate or large.

Our univariate tree structured GARCH procedure has a straightforward application in multivariate models where the conditional variance of an individual series is modeled as a function of the individual lagged values and individual lagged volatility. The multivariate cross-dependence is then modeled with cross-dependent innovations. For example, individual tree GARCH models for volatilities lead to an attractive version of the multivariate, constant conditional correlation model (Bollerslev, 1990). Another straightforward extension of our methodology is tree structured GARCH(p, q) modeling with $p > 1$ and/or $q > 1$. As already mentioned in section 1, this may be of minor importance since the general model in (1.1) is in vogue and believed to capture the most important aspects of the underlying mechanism.

Rigorous statistical inference is difficult due to the nature of non-continuity with thresholds or trees. For example, if the underlying conditional variance function $f(\cdot,\cdot)$ in (2.1) is sufficiently smooth but our threshold model is fitted as an approximation, we conjecture that the (first) fitted threshold estimate has convergence rate $n^{-1/3}$ with non-normal limiting distribution: this may be shown using the empirical process results from Kim and Pollard (1990). If the aim is primarily to construct better volatility forecasts, which in turn can be used for dynamic risk management (cf. McNeil and Frey, 2000), we choose the route to select a model in terms of an information/complexity criterion rather than the somewhat inappropriate structural tool of testing. As a simple solution, we use the AIC criterion: we believe and have demonstrated on numerical examples, that it can be used as a reasonable guideline.

References

[1] Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. J. of Econometrics **31**, 307–327.

[2] Bollerslev, T. (1990). Modelling the coherence in short-run nominal exchange rates: a multivariate generalized ARCH model. The Review of Economics and Statistics **72**, 498–505.

[3] Breiman, L., Friedman, J.H., Olshen, R.A. & Stone, C.J. (1984). *Classification and Regression Trees*. Wadsworth, Belmont (CA).

[4] Bühlmann, P. and McNeil, A.J. (1999). Nonparametric GARCH models. Preprint, ETH Zürich.

[5] Clark, L.A. and Pregibon, D. (1993). Tree-based models. In *Statistical Models in S*, pp.377–419, auths. J.M. Chambers and T.J. Hastie. Chapman & Hall, London.

[6] Hafner, C.M. (1998). Estimating high-frequency foreign exchange rate volatility with nonparametric ARCH models. J. of Statistical Planning and Inference **68**, 247–269.

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

[8] Kim, J. and Pollard, D. (1990). Cube root asymptotics. Annals of Statistics **18**, 191–219.

[9] McNeil, A.J. and Frey, R. (2000). Estimation of tail-related risk measures for heteroscedastic financial time series: an extreme value approach. To appear in J. of Empirical Finance.

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

[11] Rabemananjara, R. and Zakoian, J.M. (1993). Threshold ARCH models and asymmetries in volatility. J. of Applied Econometrics **8**, 31–49.

[12] Tong, H. (1990). *Non-linear Time Series. A Dynamical System Approach.* Oxford University Press.

[13] Yang, L., Härdle, W. and Nielson, J.P. (1999). Nonparametric autoregression with multiplicative volatility and additive mean. J. of Time Series Analysis **20**, 579–604.

[14] Zumbach, G. (1999). The pitfalls in fitting GARCH(1,1) processes. Preprint. Olsen & Associates, Zürich.

Appendix

Estimation with log-transform and generalized additive model (GAM).
A nonparametric estimate for σ_t^2 can be derived as follows.
1. Estimate the conditional mean $\mu_t = \mathbb{E}[X_t|\mathcal{F}_{t-1}]$ by a GAM model,

$$\hat{\mu}_t = \hat{g}_1(X_{t-1}) + \hat{g}_2(X_{t-2}),\ t = 3, \ldots, n,$$

with nonparametric estimates $\hat{g}_i(\cdot)$ obtained from a least squares backfitting algorithm, cf. Hastie and Tibshirani (1990).
2. Compute $Y_t = \log((X_t - \hat{\mu}_t)^2),\ t = 3, \ldots, n$.
3. In model (2.1), but with more general additive μ_t instead of ϕX_{t-1}, we have $Y_t \approx \beta + \log(\sigma_t^2) + (\log(Z_t^2) - \beta)$ with $\beta = \mathbb{E}[\log(Z_t^2)]$. Denote by $\gamma_t = \beta + \log(\sigma_t^2)$. Fit a GAM model with the transformed data Y_3^n,

$$\hat{\gamma}_t = \hat{h}_1(X_{t-1}) + \hat{h}_2(X_{t-2}),\ t = 3, \ldots, n,$$

with nonparametric estimates $\hat{h}_i(\cdot)$ obtained from a least squares backfitting algorithm with response variables Y_t, cf. Hastie and Tibshirani (1990).

4. Back-transform $\delta_t = \exp(\hat{\gamma}_t) \approx \exp(\beta)\sigma_t^2 = \frac{1}{c}\sigma_t^2$ and build $R_t^2 = (X_t - \hat{\mu}_t)^2/\delta_t \approx c\ Z_t^2$. Thus, set

$$\hat{c} = (n)^{-1} \sum_{t=1}^{n} R_t^2.$$

5. Then, set

$$\hat{\sigma}_t^2 = \hat{c}\delta_t.$$

6. Iterate steps 1.-5. Thereby use weighted estimation in step 1. with weights $w_t = \frac{1}{\hat{\sigma}_t}$, where $\hat{\sigma}_t^2$ is the estimate from the previous iteration step. Stop iterating by checking convergence of $\hat{\sigma}_t^2$ and $\hat{\mu}_t$.

A related technique is given in Yang et al. (1999): they don't use the log-transform but work with the squared observations and dependent, uncorrelated innovations (even when not estimated).

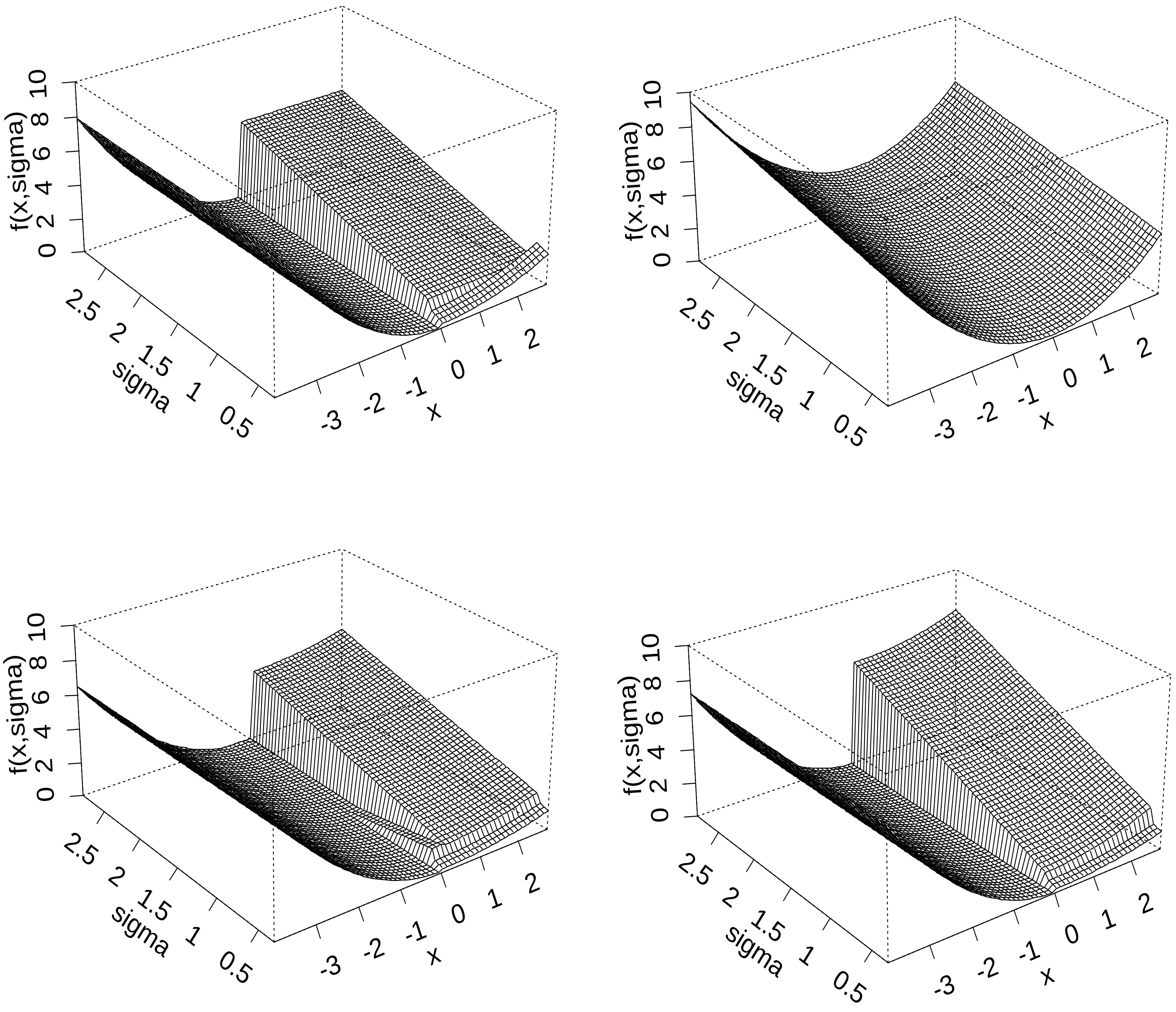

Figure 3.1: Top left: true conditional variance $f(x, \sigma)$ given by (3.1), plotted against x and σ. This is used in model (2.1) with scaled t_6-distributed innovations to simulate data (data 4 from Table 3.1) which is at the basis of the other pictures. Top right: estimated conditional variance from classical GARCH(1,1) model with scaled t_ν-distributed innovations (ν unknown). Bottom left: estimated conditional variance from tree GARCH model with standard normal innovations. Bottom right: estimated conditional variance from tree GARCH model with scaled t_ν-distributed innovations (ν unknown).

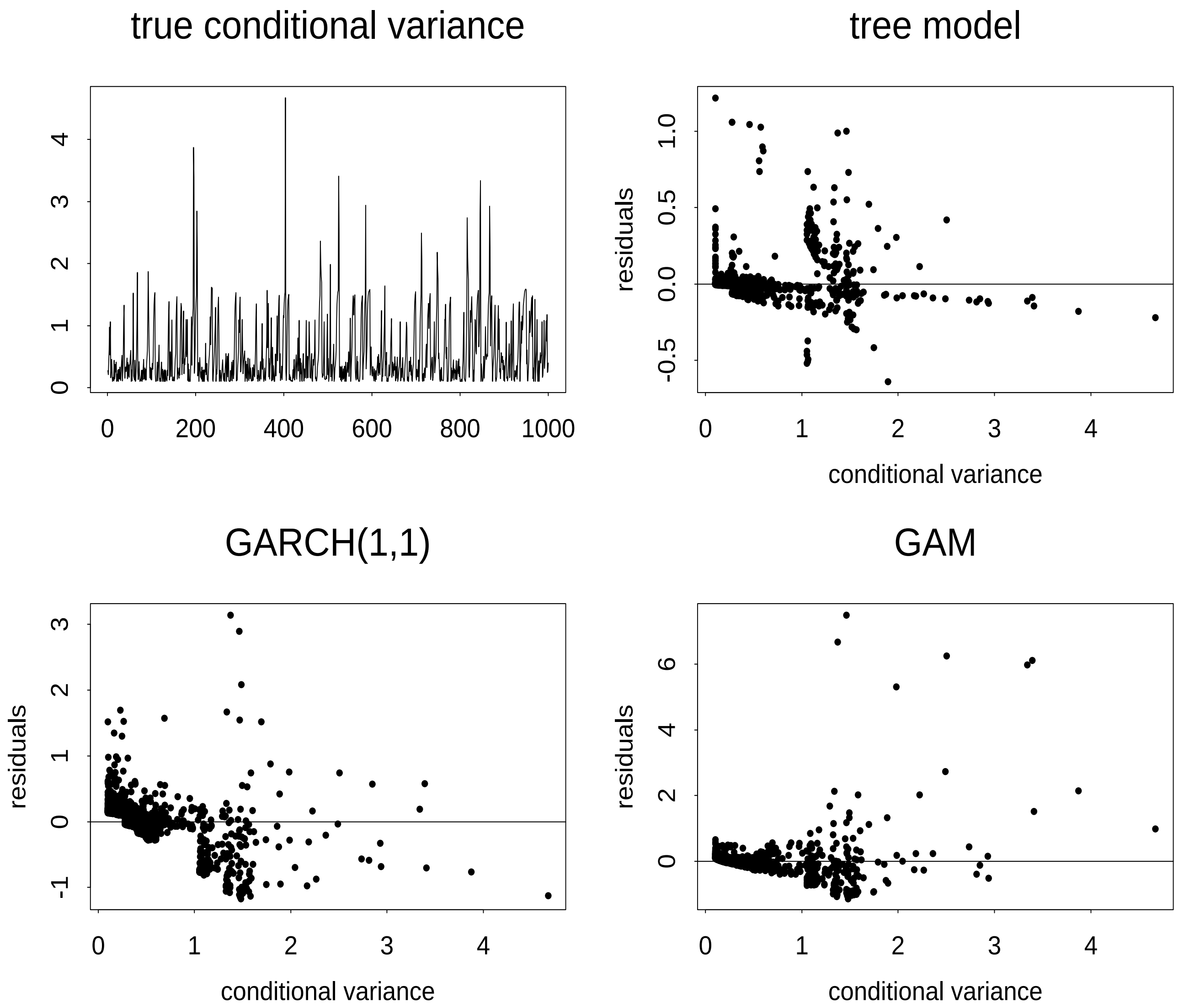

Figure 3.2: Top left: simulated volatility function from model (2.1) with (3.1) and scaled t_6-distributed innovations (data 4 from Table 3.1). Top right: residuals $\hat{Z}_t$ from tree GARCH with scaled t_ν-distributed innovations (ν unknown) against true conditional variance σ_t^2. Bottom left: residuals $\hat{Z}_t$ from classical GARCH(1,1) with scaled t_ν-distributed innovations (ν unknown) against true conditional variance σ_t^2. Bottom right: residuals $\hat{Z}_t$ from GAM model (as described in the Appendix) against true conditional variance σ_t^2.

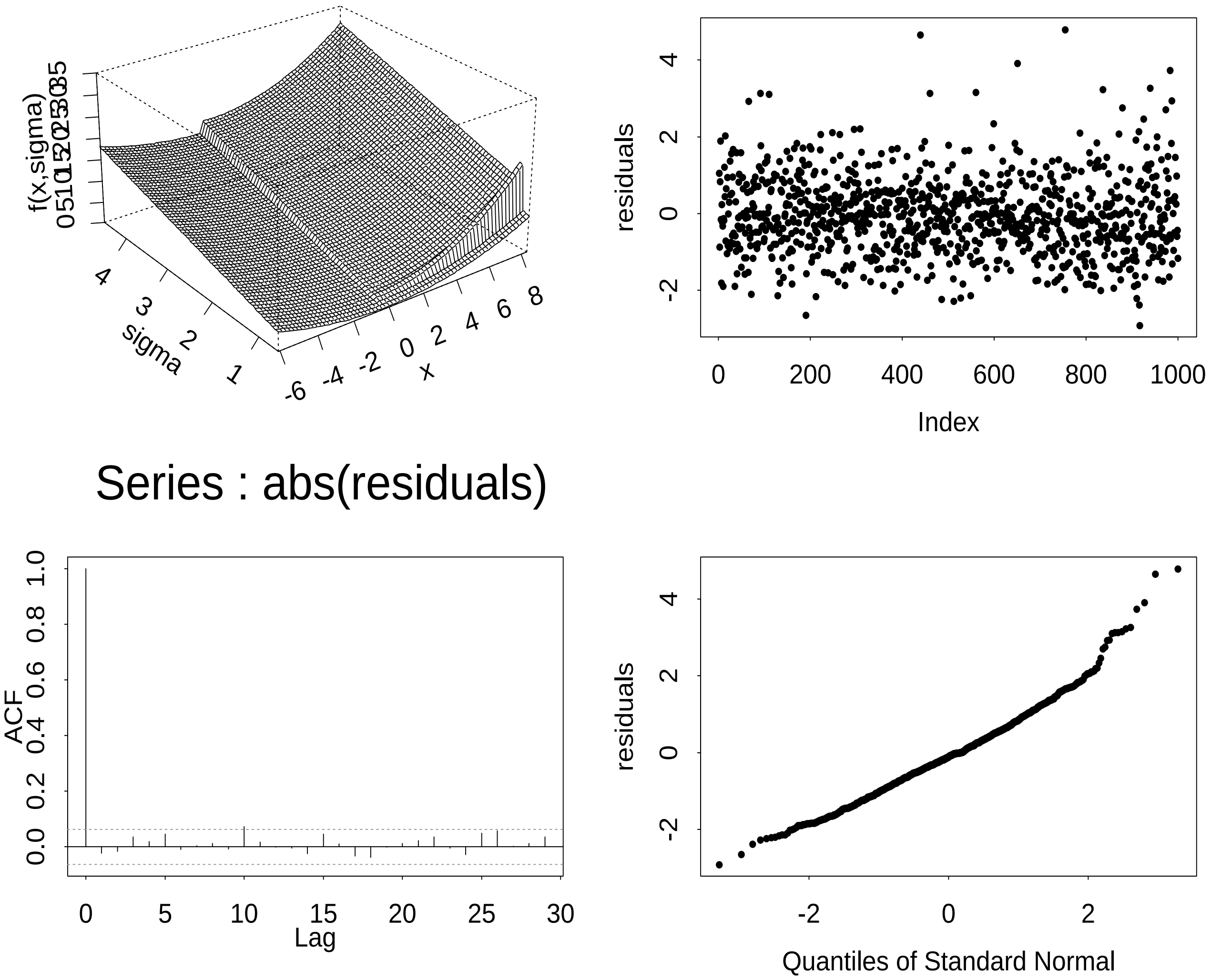

Figure 3.3: Results for negative log-returns of the DAX index using tree structured GARCH with $\mathcal{N}(0,1)$-distributed innovations. Top left: estimated function $f(x,\sigma)$ for the conditional variance, plotted against x and σ. Top right: residuals $\hat{Z}_t = (X_t - \widehat{\mu}_t)/\widehat{\sigma}_t$ from the tree GARCH model against time. Bottom left and right: autocorrelation function of the absolute residuals $|\hat{Z}_t|$ and normal-plot for the residuals $\hat{Z}_t$, respectively.

	Tree structured GARCH					
	Thresholds	AIC	L_1	L_2	OS-L_1	OS-L_2
data 1	$\widehat{d}_1$ = -0.044736 in x $\widehat{d}_2$ = 0.773178 in σ^2 $\widehat{d}_3$ = 0.556751 in σ^2	1842.201	102.8911	43.47956	0.044393	0.061369
data 2	$\widehat{d}_1$ = -0.016448 in x $\widehat{d}_2$ = 0.468750 in σ^2 $\widehat{d}_3$ = 0.724830 in x	1862.673	74.45144	25.22954	0.028189	0.043028
data 3	$\widehat{d}_1$ = -0.008402 in x $\widehat{d}_2$ = 0.188899 in x $\widehat{d}_3$ = 0.981620 in σ^2	1885.969	101.6080	47.08707	0.032485	0.031135
average	–	1863.614	92.98351	38.59872	0.035022	0.045177
data 4	$\widehat{d}_1$ = -0.008463 in x $\widehat{d}_2$ = 0.385467 in σ^2 $\widehat{d}_3$ = 0.313737 in x	1743.976	114.0162	40.50913	0.056668	0.032445
	$\widehat{d}_1$ = -0.008463 in x $\widehat{d}_2$ = 0.406501 in σ^2	1709.942	99.52060	50.98907	0.092520	0.020926

	GARCH(1,1)					GAM	
	AIC	L_1	L_2	OS-L_1	OS-L_2	L_1	L_2
data 1	2009.407	225.6830	116.2419	0.150125	0.175476	210.2130	347.0649
data 2	2078.053	261.5570	159.4396	0.476803	0.590879	235.0034	357.5037
data 3	2073.761	284.3699	227.1664	0.140545	0.181043	294.8932	443.7396
average	2053.7403	257.2033	167.6160	0.255824	0.315799	246.7032	382.7694
data 4	1897.215	257.8995	154.1101	0.263674	0.111068	211.4139	253.5949
	1818.409	265.6812	165.3186	0.324841	0.119131		

Table 3.1: Estimated thresholds $\hat{d}_i$ and goodness of fit measures for four independent simulations of model (2.1) with (3.1): with standard normal innovations (data 1, data 2, data 3) and scaled t_6 innovations (data 4). The likelihood for estimation is based on standard normal innovations (data 1, data 2, data 3); for data 4, we use standard normal innovation likelihood (upper part) and scaled t_ν innovation likelihood with ν unknown (lower part).

	Tree structured GARCH			GARCH(1,1)	
	Thresholds	AIC	IS-L$_2$	AIC	IS-L$_2$
DAX	$\widehat{d}_1$ = -0.354193 in x $\widehat{d}_2$ = 1.235410 in σ^2 $\widehat{d}_3$ = 1.657356 in σ^2	2776.238	8555.143	2785.297	9309.807
	$\widehat{d}_1$ = -0.354193 in x $\widehat{d}_2$ = -0.889087 in x $\widehat{d}_3$ = 1.657356 in σ^2 $\widehat{d}_4$ = -0.508199 in x	2764.430	8507.640		
BMW	$\widehat{d}_1$ = -0.321663 in x $\widehat{d}_2$ = 1.110003 in σ^2	3155.012	12059.22	3165.068	12063.92

Table 3.2: Estimated thresholds $\hat{d}_i$, the AIC-statistic and the in-sample L_2-loss IS-L$_2$ for negative log-returns of the DAX index and the BMW stock price. The tree GARCH model (with $\mathcal{N}(0,1)$-distributed innovations) is fitted with mesh = 8 (DAX, upper part; BMW) and mesh = 16 (DAX, lower part).