Online Monitoring with Local Smoothing Methods and Adaptive Ridging

Jochen Einbeck[*] and Göran Kauermann[†]

18th November 2002

Abstract

We consider online monitoring of sequentially arising data as e.g. met in clinical information systems. The general focus thereby is to detect breakpoints, i.e. timepoints where the measurement series suddenly changes the general level. The method suggested is based on local estimation. In particular, local linear smoothing is combined by ridging with local constant smoothing. The procedure is demonstrated by examples and compared with other available online monitoring routines.

Key Words: Breakpoint Detection, Online Monitoring, Local Linear Smoothing, Ridging.

[*] Institut für Statistik, Ludwig-Maximilians-Universität München, Akademiestraße 1, 80799 München, Germany

[†] Department of Statistics and Robertson Centre, University of Glasgow, Boyd Orr Building, Glasgow G12 8QQ, United Kingdom

1 Introduction

A considerable number of papers in the last years focussed on modelling and testing of edges and jumps in smooth functions, see e.g. McDonald & Owen (1986), Hall & Titterington (1992), Chu, Glad, Godtliebsen & Marron (1998), Müller & Stadtmüller (1999). These methods are however preferably or exclusively designed for data which are analyzed "offline". This means the entire data set is available for the analysis. In contrast, "online" monitoring is required if observations arrive successively in time. Then at each time point a decision is required whether a jump or edge has occurred. In this paper we will extend some of the "offline" tools above for monitoring data online. We develop an online test checking for breakpoints.

The analysis of data occurring online is an important issue in various fields of science and industry. This includes quality control management, time series in finance or online monitoring of clinical information systems. A general overview of existing procedures for online monitoring is found in Basseville & Nikiforov (1993). The use of online methods in clinical information systems has been focussed by e.g. by Daumer & Falk (1998), who make use of a Kalman filter to detect jumps and thresholds in the (online) ECG profile of a patient after surgery. Imhoff & Bauer (1996) and Bauer, Gather & Imhoff (1999) make use of a time series approach for online monitoring while Daumer (1997) uses an adaptive control chart based on moving averages. In all these papers the general focus is to detect sudden structural changes in order to give alarm.

The general problem for online monitoring we are considering here can be described as follows. Assume that at time-point t the measurement y_t is observed. It is assumed that y_t follows the stochastic model

$$y_t = \mu(t) + \varepsilon_t \tag{1}$$

where $\mu(t)$ is the mean function in time, which possibly also depends on other covariates, and ε_t is a random noise, which is allowed to be correlated with previous observations. Both, y_t and hence ε_t are allowed to be multivariate, but we restrict to the univariate case here. Based on the information available at time-point t, i.e. based on $y_1, \ldots, y_t$, it is to decide whether $\mu(t)$ has a breakpoint at time-point t. A breakpoint here means that $\mu(t)$ is discontinuous, i.e. there is a jump at t, or $\mu(t)$ has a discontinuous first derivative, i.e. there is an edge or sharp bend at t. Online monitoring of the data should give alarm if a breakpoint occurs at time-point t.

A convenient approach is to compare the observed value y_t with a predictor $\widehat{y}_t$. Alarm is given if y_t differs from the predictor by more than the threshold A_t, say, i.e. if

$$|y_t - \widehat{y}_t| > A_t. \tag{2}$$

The threshold A_t is thereby chosen such that sensitivity of the alarm rule is achieved while the probability of false alarms is small. The prediction $\widehat{y}_t$ is calculated from previous values $y_{t-h}, \ldots y_{t-1}$, with h as time lag. Daumer (1999) suggests to calculate $\widehat{y}_t$ by a running mean calculated from $y_{t-h}, \ldots y_{t-d}$, where d is a second time lag with $1 < d < h$. Hence observations in the near past are left unconsidered. The time lag d serves as delay for the running mean and Daumer shows that for $d > 1$ the alarm rule (2) improves its performance compared to taking $d = 1$. In this paper we apply more sophisticated smoothing techniques instead of a simple running mean. We make use of local polynomial fitting (see e.g. Fan & Gijbels, 1996) which reacts better on structural changes and moreover can cope for smooth shifts, unlike the running mean.

Considering (2) it becomes obvious, that the alarm rule basically depends on the value of y_t. This in turn implies a high variance of the procedure. We therefore replace y_t in (2) by a smooth estimate of $\mu(t)$. In the same way we replace the predictor by a second smooth estimate. This means we consider the alarm rule

$$|\widehat{\mu}_1(t) - \widehat{\mu}_2(t)| \quad > \quad A_t \tag{3}$$

where $\widehat{\mu}_1(t)$ and $\widehat{\mu}_2(t)$ are two estimates of $\mu(t)$. The first estimate $\widehat{\mu}_1(t)$ is thereby calculated as long term estimate from $y_{t-h_1}, \ldots, y_t$ while $\widehat{\mu}_2(t)$ is a short term smoother obtained from $y_{t-h_2}, \ldots, y_t$, where $h_2 < h_1$. The major difference of (3) compared to (2) is, that we do not compare the current observation with its predictor, but we compare two estimates of the mean function. The basic idea behind this is that if $\mu(t)$ has a jump or a sharp bend at t, the long term estimate $\widehat{\mu}_1(t)$ and the short term estimate $\widehat{\mu}_2(t)$ will essentially differ. If in contrast $\mu(t)$ is smooth, both smooth estimates will basically be the same. Hence the alarm rule (3) can be seen as smooth test statistic, where large values indicate a violation in the smoothness of $\mu(t)$.

The bandwidth h_2 which is chosen for the short term estimate mainly determines the speed of reaction of the alarm. Taking a large value for h_2, the reaction time and the specitivity of the alarm rule increases while the variance of the alarm rule decreases so that false alarms are less probable. Using a small bandwidth h_2 on the other hand improves the reaction time and the sensitivity of the alarm rule (3) but the variability increases. The second tuning parameter h_1 decides in which depth the method is searching for breakpoints. For small values of h_1 mainly short term changes will be detected, while with a large value of h_1 the focus is on detecting long term breaks of the structure of the time series. Beside the choice of the two bandwidths h_1 and h_2 the fixing of the threshold A_t is required which however results from simple variance calculations.

The choice of the applied smoothing method is thereby essential. Generally speaking, smoothing methods are weak in detecting jumps since they smooth over edges or jumps. Once a jump occurs and is detected, it is therefore necessary that the smooth estimates adjust quickly for the new level or shift. It is well known that local linear smoothing and local constant smoothing, which is a simple running mean, react quite differently at the boundary of the support points. Note that by definition, the online estimates are calculated at the boundary. We will combine both estimates using a ridge regression, as suggested in Seifert & Gasser (2000) for “offline” analysis. The ridge regressor thereby results as weighted mean of the local linear and the local constant estimate.

2 Local Linear Smoothing and Breakpoint Detection

We calculate the long term estimate by fitting a local linear model to the data pairs $(t-i, y_{t-i})$ for $i = 0, 1, \ldots, h_1$. Let therefore $K_1(\cdot)$ denote a kernel function with support $[0, h_1]$. An example is found by taking $K_1(\cdot)$ as the truncated normal density with mean $h_1/2$ and variance $(h_1/4)^2$. The estimate $\widehat{\mu}_1(t)$ is then obtained by fitting a weighted linear model using the kernel $K_1(\cdot)$ as weight function. It is not difficult to show that the resulting estimate is the weighted mean

$$\widehat{\mu}_1(t) \quad = \quad \sum_{i=0}^{h_1} w_{i,1} y_{t-i} \tag{4}$$

with weights

$$\begin{aligned} w_{i,1} &= (1,0) K_1(i) \left(\sum_{j=0}^{h_1} K_1(j) \begin{pmatrix} 1 \\ -j \end{pmatrix} (1, -j) \right)^{-1} \begin{pmatrix} 1 \\ -i \end{pmatrix} \\ &= \frac{K_1(i)(S_{h_1,2} + i S_{h_1,1})}{S_{h_1,0} S_{h_1,2} - S_{h_1,1}^2} \end{aligned} \tag{5}$$

where $S_{h_1,j} = \sum_{i=0}^{h_1} K_1(i)(-i)^j$ for $j = 0, 1, 2$. It is important to note that the weights do not change in t and hence they can be calculated once and no updating is required.

In the same fashion one obtains the short term estimate $\widehat{\mu}_2(t)$ as local linear fit to the data $(t-i, y_{t-i})$, $i = 0, \ldots, h_2$. Let therefore $K_2(\cdot)$ be a kernel density with support $[0, h_2]$, e.g. a half sided normal distribution.

Our experience is that the particular choice of the kernel functions K_1 and K_2 is not very crucial as long as they follow the setting shown in Figure 1 and are not truncated too roughly on the left hand side. For example, setting $K_1(\cdot)$ as the uniform kernel with $K_1(x) = 1/h_1$ for $x \in [0, h_1]$ might cause an artificial alarm signal when the left border of the support of K_1 is passing a jump or bend which should already have been detected the time span h_1 before.

Figure 1

For $i = 0, \ldots, h_2$ we set

$$w_{i,2} \quad = \quad \frac{K_2(i)(S_{h_2,2} + iS_{h_2,1})}{S_{h_2,0}S_{h_2,2} - S_{h_2,1}^2}$$

with $S_{h_2,j} = \sum_{i=0}^{h_2} K_2(i)(-i)^j$ for $j = 0, 1, 2$, while $w_{i,2} = 0$ for $i > h_2$. The short term estimate is then available through

$$\widehat{\mu}_2(t) \quad = \quad \sum_{i=0}^{h_1} w_{i,2} y_{t-i} = \sum_{i=0}^{h_2} w_{i,2} y_{t-i}. \tag{6}$$

The weights are for convenience constructed such that the vectors $\boldsymbol{w}_1 = (w_{0,1}, \ldots, w_{h_1,1})^T$ and $\boldsymbol{w}_2 = (w_{0,2}, \ldots, w_{h_1,2})^T$ have equal length. We now combine the two estimates in the alarm rule (3). If $\mu(t)$ is smooth in $[t - h_1, t]$ the bias of $\widehat{\mu}_1(t) - \widehat{\mu}_2(t)$ can be approximated by

$$\begin{aligned} E\{\widehat{\mu}_1(t) - \widehat{\mu}_2(t)\} \quad &= \quad \mu''(t)/2 \sum_{i=0}^{h_1} (w_{i,1} - w_{i,2})(-i)^2 + \ldots \\ &= \quad \mu''(t)/2 \left(\frac{S_{h_1,2}^2 - S_{h_1,1}S_{h_1,3}}{S_{h_1,0}S_{h_1,2} - S_{h_1,1}^2} - \frac{S_{h_2,2}^2 - S_{h_2,1}S_{h_2,3}}{S_{h_2,0}S_{h_2,2} - S_{h_2,1}^2} \right) + \ldots . \end{aligned} \tag{7}$$

The approximation is based on a simple Taylor series and is heuristic in nature. Rigorous quantification of the bias would require a number of assumptions to hold, most of which are not met in practice. For instance in standard smoothing literature theoretical developments are based on the assumption that values t are getting infinitely dense. In our online scenario however we assume that time t is realised on an equidistant grid. For this reason we do not investigate the bias from a theoretical point of view. However, considering (7) shows that the bias gets large if $\mu''(t)$ is large, which is the case if $\mu(\cdot)$ rapidly changes its direction at t. As extreme case this results in a jump or sharp bend. The quantity $\widehat{\mu}_1(t) - \widehat{\mu}_2(t)$ in the alarm rule (3) can therefore be seen as an empirical estimate for the second order derivative of $\mu(\cdot)$. If the resulting value is large in absolute terms the resulting function is likely to be rough or unsmooth in t.

The choice of the threshold A_t in (3) requires the estimation of the variability of $\widehat{\mu}_1(t) - \widehat{\mu}_2(t)$. We rewrite $\widehat{\mu}_1(t) - \widehat{\mu}_2(t)$ as

$$\widehat{\mu}_1(t) - \widehat{\mu}_2(t) = \sum_{i=0}^{h_1} w_i y_{t-i} \tag{8}$$

where $w_i = w_{i,1} - w_{i,2}$. Assuming local stationarity, simple calculation leads to

$$var\{\widehat{\mu}_1(t) - \widehat{\mu}_2(t)\} \quad = \quad \sum_{i=0}^{h_1} w_i^2 \gamma(0) + 2 \sum_{i=0}^{h_1} \sum_{j>i}^{h_1} w_i w_j \gamma(i-j)$$

where $\gamma(d) = cov(y_{l-d}, y_l)$ is the covariance function and $\gamma(0) = var(y_l)$ with $l = t - h_1, \ldots, t$.

Estimation of (9) can then be done by the simple moment based estimate (see Brockwell & Davis, 1987)

$$\widehat{\gamma}(d) = \frac{c_d}{h+1-d} \sum_{i=t-h}^{t-d} \{y_i - \widehat{\mu}_2(i)\}\{y_{i+d} - \widehat{\mu}_2(i+d)\}. \tag{9}$$

where $h > h_1$ is some timelag expressing the local stationarity of the process. In the following we provide some heuristics to find a suitable value of c_d. Assuming y_l, $l = 1, 2, \ldots$ to be independent one finds for $d = 0$ in (9) by taking expectation

$$E\left[\sum_{i=t-h}^{t} \{y_i - \widehat{\mu}_2(i)\}^2\right] \quad = \quad \gamma(0)(h+1)\left(1 - 2w_{0,2} + \sum_{j=0}^{h_2} w_{j,2}^2\right).$$

Skipping the assumption of independence, one gets for $0 < d \leq h_2$

$$E\left[\sum_{i=t-h}^{t-d} \{y_i - \widehat{\mu}_2(i)\}\{y_{i+d} - \widehat{\mu}_2(i+d)\}\right] \quad = \quad (h+1-d)\left[\gamma(d)\left(1 - 2w_{0,2} + \sum_{j=0}^{h_2} w_{j,2}^2\right) + \ldots\right]$$

with ... standing for a collection of terms build from $\gamma(i)$, $i \neq d$. Detailed consideration shows that the terms not explicitly listed are of order $1/h_2$ and for simplicity of calculations they are neglected subsequently. This suggests to set $c_d = 1/(1 - 2w_{0,2} + \sum_{j=0}^{h_2} w_{j,2}^2)$ for all $d = 0, \ldots, h_2$, to achieve a bias reduced variance estimate. Usually the constant c_d obtained in this manner is slightly bigger than 1. For $d > h_2$, we suggest to set $c_d = 0$ and thus $\gamma(d) = 0$.

The computation of (9) in every timepoint can be accelerated by making use of the following iterative update scheme. Let $\boldsymbol{d}_{t,h} = \{y_t - \widehat{\mu}_2(t), y_{t-1} - \widehat{\mu}_2(t-1), \ldots, y_{t-h} - \widehat{\mu}_2(t-h)\}^T$ and

$$\boldsymbol{D}_{t,h} \quad = \quad \begin{pmatrix} \frac{\boldsymbol{d}_{t,h}}{h+1} & \begin{matrix} \mathbf{0}_1 \\ \frac{\boldsymbol{d}_{t,h-1}}{h} \end{matrix} & \begin{matrix} \cdots \\ \cdots \end{matrix} & \begin{matrix} \mathbf{0}_{h_1} \\ \frac{\boldsymbol{d}_{t,h-h_1}}{h-h_1+1} \end{matrix} \end{pmatrix}$$

where $\mathbf{0}_d$ are column vectors of zeros with length d. The covariance vector at time point t can then be estimated by $\widehat{\boldsymbol{\gamma}_t} = \boldsymbol{d}_{t,h}^T \boldsymbol{D}_{t,h} \boldsymbol{C}$, where $\widehat{\boldsymbol{\gamma}_t} = \{\widehat{\gamma}_t(0), \ldots, \widehat{\gamma}_t(h_1)\}$, $\boldsymbol{C} = \mathrm{diag}(c_i)_{0 \leq i \leq h_1}$ and the subscript t indicates that information available at timepoint t is used. Simple matrix algebra (see appendix) provides the approximative recursive formula

$$\widehat{\boldsymbol{\gamma}}_{t+1} \quad \approx \quad \frac{1}{h+1}(y_{t+1} - \widehat{\mu}_2(t+1)) \boldsymbol{d}_{t+1,h_1}^T \boldsymbol{C} + \frac{h}{h+1} \widehat{\boldsymbol{\gamma}}_t. \qquad (10)$$

Defining the covariance matrix $\boldsymbol{\Gamma} = [\boldsymbol{\Gamma}]_{ij} = [\gamma(|i-j|)]_{ij}$ for $i, j = 0, \ldots, h_1$ one gets the variance estimate

$$\widehat{\mathrm{var}}(\widehat{\mu}_1(t) - \widehat{\mu}_2(t)) \quad = \quad \boldsymbol{w} \widehat{\boldsymbol{\Gamma}} \boldsymbol{w}^T \qquad (11)$$

where $\boldsymbol{w} = (w_0, \ldots w_{h_1})$ and $\widehat{\boldsymbol{\Gamma}}$ is a plug in estimate of $\boldsymbol{\Gamma}$. This suggests the alarm threshold

$$A_t = a \sqrt{\widehat{\mathrm{var}}(\widehat{\mu}_1(t) - \widehat{\mu}_2(t))} \qquad (12)$$

where a is chosen such that the alarm rule is sensitive and false alarms are less probable. This provides a simple test on presence of a breakpoint: We reject the hypothesis “No breakpoint at time t” if

$$|T_t| = \left| \frac{\widehat{\mu}_1(t) - \widehat{\mu}_2(t)}{\sqrt{\widehat{\mathrm{var}}(\widehat{\mu}_1(t) - \widehat{\mu}_2(t))}} \right| > u_{1-\frac{\alpha}{2}} \qquad (13)$$

where α is the error probability and $u_{1-\frac{\alpha}{2}}$ is the $1 - \alpha/2$ quantile of the $N(0, 1)$-distribution.

3 Practical Adjustments

3.1 Ridging

In Section 2 we suggested to use local linear fitting to calculate the long and short term estimates. All estimates are calculated at the boundary, where local polynomial smoothers are known to be more variable than local constant smoothers. In terms of variability one therefore has to consider the Nadaraya-Watson estimate

$$\widehat{\mu}_{1,NW}(t) \;\; = \;\; \sum_{i=0}^{h_1} w_{i,1,NW} y_{t-i} \qquad (14)$$

with $w_{i,1,NW} = K_1(i)/S_{h_1,0}$ as a competitor to $\widehat{\mu}_1(t)$. Figure 2

Figure 2 shows the behavior of the local estimates when used with the alarm rule (3) for independent Gaussian errors. Both estimates detect the jump at 200 and the bend at 400, but the bend at 600 is only found by the local linear fit, since this adopts the inclination. Hence, one should use a local linear fit when there is a slope in the data while local constant appears more appropriate if the data are flat. Considering the local linear fit in more depth uncovers a further drawback. The local linear fit adjusts for the model violation shortly after the jump, while the local constant fit reacts delayed. Thereafter however the local linear fit over-steers the shift and the local constant gets superior. Figure 3 gives a tutorial to demonstrate this point. In order to balance local linear and local constant fitting we propose to Figure 3 use ridging as suggested in Seifert & Gasser (2000). This means we replace the long term estimate by

$$\widehat{\mu}_{1,ridge}(t) = \lambda_t \widehat{\mu}_{1,NW}(t) + (1-\lambda_t)\widehat{\mu}_1(t) \tag{15}$$

where $\lambda_t \in [0,1]$ is the ridge parameter. The ridge estimate again results as a weighted sum of the observations y_i so that variance calculations for the ridge estimate are straight forward.

The ridge parameter λ_t in (15) is allowed to depend on time t. Seifert & Gasser (2000) suggest a rule of thumb to use design adaptive ridging. This is of little use for our scenario since the design is fixed and regular and observations are recorded at equidistant time intervals. For online monitoring it is more reasonable to use data adaptive ridging by considering the shape of the mean function $\mu(t)$. The general idea is to work with local constant smoothers if the mean is constant while local linear smoothing should be used if there is a drift. We incorporate this by estimating the slope of $\mu(t)$ via the local linear estimate

$$\widehat{\beta}(t) = \sum_{i=0}^{h_1} v_i y_{t-i}$$

with $v_i = K_1(i)(S_{h_1,1} + iS_{h_1,0})/(S_{h_1,1}^2 - S_{h_1,0}S_{h_1,2})$. The principle is now that small squared slope estimates $\widehat{\beta}(t)$ should lead to local constant fitting, i.e. large values of λ_t. We achieve this by setting

$$\lambda_t = e^{-c\widehat{\beta}^2(t) \qquad (c>0)}. \tag{16}$$

Setting the parameter c equal to zero leads to local constant fitting while $c \to \infty$ gives local linear smoothing. In Figure 2 we use $c = 50$. It becomes obvious that the ridge estimate combines the advantages of local linear and local constant fitting. Figure 4 shows the value of λ_t over time in this example. We pick up the task of how to select the constant c at the end of example 4.2. Figure 4

Criterion (16) is a suggestion but in no way unique. Other choices for choosing λ_t can easily be constructed. We experimented with the Seifert & Gasser suggestion of design adaptive ridging. Not surprisingly this did not convince since it led to constant ridging, i.e. non adaptive and independent of t. To let the ridging parameter depend on the slope any monotonically decreasing function in $|\beta(t)|$ would work. The chosen form (16) however proved to work satisfactory in practice, in particular by applying the squared slope a clear distinction between the states $\lambda_t = 0$ and $\lambda_t = 1$ can be made.

3.2 Choice of h, h_1 and h_2.

In this section we will give some guidelines concerning the choice of the window sizes h, h_1 and h_2. The major importance of them has h_1, since this constant defines if breaks of short- or long term trends shall be detected. Thus what will be chosen firstly is h_1, and the other constants will be selected according to this choice.

For the selection of h_1 we provide the following rule of thumb: If the main focus is to detect breaks of trends with length down to a value D, one has to choose $h_1 \approx D$. We illustrate this point in an example: We simulated a time series of length 500 from the AR(2) process $Y_t = 0.55Y_{t-1} + 0.45Y_{t-2} + \epsilon_t$, where $\epsilon_t \sim N(0, 0.3^2)$. In Fig. 5 (top) we perform alarm detection with $h_1 = 120$, $h_2 = 25$, $h = 200$. Apart from some alarm signals in the warming-up-period in the beginning, only breaks of long term trends at $t = 160, 237, 339$ and 386 (there a long term falling trend starting at about $t = 205$ is broken) are detected. Figure 5

However, setting $h_1 = 60$, $h_2 = 15$ and $h = 100$ yields a very different picture: Now a large amount of breaks of short term trends are detected, like demonstrated in Fig. 5 (bottom). Thus, the suitable value of h_1 depends less on the data than on the intentions of the data analyst.

What concerns the choice of h_2, we already mentioned that this value influences the speed of the detection, in the sense that small values of h_2 lead to a short reaction time, but to a high variance of the procedure. From our experience, we suggest to set $h_2 \approx h_1/5$, but not smaller than 15.

We already stated that the window size h, responsible for the amount of data used to estimate the autocorrelation function, is reflecting the local stationarity of the process, i.e. by imposing a certain value of h we assume that the process is more or less stationary over a distance h. Therefore h may not be too big, to retain sufficient flexibility of the variance estimation, but should be bigger than h_1 to get reliable results. We suggest to set $h \lesssim 2 \cdot h_1$.

3.3 Missing Values and Outliers

In practical applications one is often faced with outliers or missing data which disturb the performance of the alarm rule. We suggest the following adjustments. If observation y_t is missing or outlying we impute a predicted value $\widehat{y}_t$ calculated from the previous observations. A simple setting is to use $\widehat{y}_t = \widehat{\mu}_2(t-1)$. This setting works fine to overcome both the missing values in Example 4.1 as well as the artificial outlier in Example 4.2.

In the presence of sloping data the method can be more sophisticated to cover possible shifts. We therefore predict y_t by using a linear extrapolation from the previous short term estimates via $\widehat{y}_t = \sum_{i=1}^{h_2} \nu_i \widehat{\mu}_2(t-i)$. The weights ν_i for this extrapolation can be calculated like $w_{i,2}$ in Section 2, but applying values $S_{h_2,j} (j = 0, 1, 2)$ constructed by sums starting at $i = 1$ instead of $i = 0$. These weights have to be calculated only once, so that extrapolation is numerically simple.

It remains the question of how to detect outliers. An outlier is classified as a single or small group of observations which do not follow the model. A detection rule for outliers is for instance

$$|y_t - \widehat{y}_t| > k\sqrt{\widehat{\gamma}_{t-1}(0)} \tag{17}$$

where $\widehat{y}_t$ is a predictor for y_t calculated as above and k is some positive constant. In the data examples we collected good experiences with the setting $k = 10$, even though different values can be more suitable in other data scenarios. If y_t is classified as outlier, its value is substituted by its predictor. Moreover, if (17) holds for a number of consecutive time-points alarm should be given.

3.4 Variance Calculation

The moment based variance estimator described in the previous section can be inefficient if the data are uncorrelated or the errors trace from a model with parametric dependence pattern, e.g. an AR(1) process. In the first case one can set $\gamma(i) = 0$ for $i > 0$. In the latter case one could use the assumed dependence process to improve the variance estimation. For the AR(1) process for instance one fits locally the regression model

$$\varepsilon_t = \rho\varepsilon_{t-1} + \nu_t \tag{18}$$

to the residuals ε_t, where ν_t are uncorrelated white noise errors. This yields the covariance function $\gamma(d) = \sigma^2\rho^d$ for $d = 0, \ldots, h$. In practice (18) is fitted to the fitted residual $\widehat{\varepsilon}_t = y_t - \widehat{\mu_2}(t)$ and one obtains

$$\widehat{\rho} \;=\; \sum_{i=1}^{h} \widehat{\varepsilon}_{t-i}\widehat{\varepsilon}_{t-i+1} / \sum_{i=1}^{h} \widehat{\varepsilon}^2_{t-i}.$$

The coefficient $\widehat{\rho}$ can thereby again be updated recursively from previous values as shown in the appendix.

Variance calculation suffers from jumps and edges since both estimates, the short term and the long term estimate are biased at the jumps and residuals are overfitted.

It is therefore advisable to pause online updating of the variance once a jump or outlier has been detected. This means in this case one sets $\widehat{\gamma}_t = \widehat{\gamma}_{t-1}$ until the alarm is stopped.

4 Examples

4.1 Cardio Beats

In a hospital the cardio beats per minute of the mother before the confinement are monitored. It is of interest to detect sudden changes in the recorded data. Fig. 6 shows the data and the resulting short and long term estimates. We use bandwidths $h_1 = 160, h_2 = 30, h = 300$ and a ridging constant $c = 80$. Figure 6

A special property of this data set is the large amount of missing values, displayed as data points with $Y = 0$. However, the algorithm manages to outnumber these values and hence the estimated curves are not affected as seen in the first period of missing values from $t = 176$ to $t = 196$. The bottom graph in Fig. 6 shows the standardized test statistic T_t and bands given by the 99.5%- quantile of the standard normal distribution. It is seen that all jumps are detected quickly and significantly. At points 199 and 240 a shift in the level is found while at 340 the cardio beats decrease abruptly with level changes detected at 399 and 421. Afterwards the cardio beat frequency increases slowly until it reaches the plateau. The end of the increase is detected at 604.

4.2 ECG Measurements

In the second example we apply the method to data which have been previously used in Daumer & Falk (1998) for the demonstration of their online monitoring algorithm. The data are ECG measurements taken every five seconds from a patient undergoing a skin transplantation. At $t = 219$ an artificial outlier is added. Figure 7(a) and

7(b) show the long term estimates and test statistics for different settings of the ridging parameter c in (16). For all settings the breakpoints at timepoint 120 and 285 are detected. Afterwards however the estimates behave differently. For $c = 0$ one obtains a local constant estimate. This is unable to adjust for the slope and does not find the end of the slope area at 378. Afterwards the local constant detects small level changes at 454, 497, 515 and 739. On the contrary the local linear fit, obtained for $c = \infty$, gives the end of the slope area but oversteers afterwards so that some small level changes are not uncovered, but some spurious alarm signals are given. In contrast, setting the ridging parameter $c = 120$ compensates the problem of oversteering and detects both, the end of the slope area as well as the small level changes afterwards. We return to this data example in the next section and compare our method with the procedure suggested in Daumer & Falk (1998). Figure 7

Speaking more generally, we conclude that if the data describes more or less a step function and the intention is mainly to detect jumps, we recommend $c = 0$, which corresponds to a local constant long term estimator. If however mainly breaks of trends shall be detected, one should set $c = \infty$ and thus use a local linear long term estimator. Varying c between 0 and ∞ means balancing between these two goals, and the appropriate value of c depends on the kind of breakpoints which shall be detected. A general guideline or rule of thumb on how to choose c is therefore difficult. Nevertheless, due to (16) it is observed that if the process Y_t is multiplied by a constant δ, say, coefficient c should be updated to c/δ^2 to have the same amount of ridging. Hence, the choice of c depends on the overall variability of the process, which includes areas of shifts as well. We experimented a little and found that $c = 10^4/\text{var}(Y)$ is a reasonable starting point for fine tuning of c. Apparently the variance of Y is normally unknown, since we record data online. Usually, however, one should have a notion about this value which allows to set c at the starting value for further tuning. Again, the right choice of c depends in particular on the substance matter of the online monitoring, i.e. whether one is interested in detecting jumps or breaks in trends.

5 Comparison to other methods

5.1 Autoregressive models

In Gather, Bauer, Imhoff & Löhlein (1998) it is assumed that the data follow an autoregressive model. We illustrate their method at the cardio data from above. Before applying the method, we substitute the missing values by short-term-predictors. Then the data set is divided in an estimation period and a prediction period. Since the data have to be more or less stationary during the estimation period, we choose the estimation period $t = 1, \ldots, 180$. In order to obtain a nearly balanced proportion between the amount of data in the two periods we reduce the data set to the first 380 data points. Figure 8

During the estimation period the parameters for the AR-process are estimated (the AIC criterion suggested an AR(1) model). In the prediction period it is observed whether the data points are inside or outside a confidence band surrounding the predicted values. The result is shown in Fig. 8 for a 95% confidence interval. If less than five consecutive observations are outside of the confidence interval, then they are classified as outliers, whereas a level change is detected for more than five points out of the confidence interval. Thus, the jump detected at $t = 197$ is classified as a level change at $t = 201$, compared to a detection at $t = 199$ with our adaptive ridging method.

We conclude that the AR method by Gather et al. is a fast and reliable method, but is probably not suitable for every kind of data, especially data with quickly changing rising or falling trends, whereas the local estimation method we proposed in this paper adapts to a wide range of data situations.

5.2 Phase space models

Another idea of Gather, Bauer, Imhoff & Löhlein (1998) is to plot every data point against its previous data point in a phase space. We tried this method for the first 380 cardio data points (see Fig. 9). Gather et. al. move a time window of length 60 through the data and alarm is given if the next five consecutive observations are in a different region than the previous 60 ones. In the plot, the way of the data in the phase space can be followed. Starting somewhere in the left down area of the big cluster, the line climbs up to the right top edge, then falls down and turns left to the small cluster (representing the data points $t = 197, \ldots, t = 232$) and finally climbs up again to the big cluster. Every change of the cluster represents a jump in the data. This means that alarm is given at the timepoints $t = 201$ and 236, compared to alarm signals at $t = 199$ and 240 with the adaptive ridging method. Figure 9

Though this method is very useful for visualizing the structure of the data, we think it might be difficult to use it online for reliable alarm detection, especially for sloping data, where the dividing lines between single clusters become foggy.

5.3 State Space Models

Daumer & Falk (1998) and Fahrmeir & Künstler (1999) use state space models for filtering time series. A linear state space model is given by a linear observation equation

$$y_t = z_t'\beta_t + \varepsilon_t \qquad (t = 1, 2, \ldots)$$

for the observations $y_1, y_2, \ldots$ given the states $\beta_1, \beta_2, \ldots$, which is supplemented by a linear transition equation

$$\begin{aligned} \beta_t &= F_t\beta_{t-1} + v_t \qquad (t = 1, 2, \ldots) \\ \beta_0 &= a_0 + v_0 \end{aligned}$$

with Gaussian errors ε_t and v_t, nonrandom vectors $z_1, z_2, \ldots$ and transition matrices $F_1, F_2, \ldots$. This model can be solved with Kalman filters. Daumer & Falk (1998) define such a state space model for each possible location of a jump. The resulting family of models, called a multi-process model, is examined with Bayesian methods and jumps are detected by choosing the most likely model. For detecting outliers, a 2nd multi-process model has to be introduced. Daumer & Falk (1998) apply their method to the data shown in Figure 6(a) and find changepoints at t = 120, 285, 506, 752 and 821. In contrast, the local ridging method with c=120 uncovers the changepoints 120, 285, 378, 432, 512, 739 and 788. It appears that both methods uncover abrupt changes from a long term level but the local adaptive ridging method appears more flexible and gives alarm also at short term changes. Moreover both methods are equal in the speed of detection.

6 Conclusion

We showed that local smoothing methods can be used effectively for detecting jumps and bends in online monitoring. The algorithm combines the advantages of many other breakpoint detection methods: It can be used online, since only the data given until the examined time point are necessary for the estimations. Furthermore it is able to detect jumps or bends of flat and sloping trends. The method adapts to the variability of the data, which means that it will not give alarm for a small jump within highly fluctuating data, but will give alarm for the same jump for less variable data. Finally it is worth mentioning that only few computational effort is required, because the weights needed for the estimates have only to be calculated once and the variance calculations follow a simple update rule.

The only technical problem, namely the over-steering, can be solved quite satisfactory by adaptive ridging. However, we shouldn't suppress that it can't be avoided completely (see. Fig. 2 and 6). If one wants to exclude it totally, one has either to use only the data *after* a jump for the estimations of $\widehat{\mu}_1(t)$ and $\widehat{\mu}_2(t)$, which requires recalculating all weights after every jump, or to use methods like edge-preserving smoothing (see Chu, Glad, Godtliebsen & Marron, 1998). However, both ways require additional computational effort, so that it is questionable whether they convince in practice.

Acknowledgements

The authors gratefully acknowledge support in various aspects received from the Sonderforschungsbereich (SFB 386) funded by the Deutsche Forschungsgemeinschaft.

The authors are also grateful to the Institute of Medical Statistics and Epidemiology at the Technical University Munich (TUM) for providing the data and especially to Dr. Daumer for explaining aims and problems of applied online monitoring and for providing informations about the medical background of the data sets used in this paper. The data were recorded within the the project B2 of the SFB 386 in collaboration with Trium Analysis Online GmbH, Women hospital and policlinic of TUM (Prof. Dr. K. T. M. Schneider, PD Dr. J. Gnirs), Institute of anaesthesiology, TUM (Prof. Dr. E. Kochs, O. Möllenberg).

A Technical Details

Derivation of (10)

Note that

$$
\begin{aligned}
&\boldsymbol{d}_{t+1,h}^{T}\boldsymbol{D}_{t+1,h} \\
&= \left(y_{t+1}-\widehat{\mu}_2(t+1), \boldsymbol{d}_{t,h-1}^{T}\right)
\begin{pmatrix}
\frac{y_{t+1}-\widehat{\mu}_2(t+1)}{h+1} & \mathbf{0}_1 & \cdots & \mathbf{0}_{h_1} \\
\frac{\boldsymbol{d}_{t,h-1}}{h+1} & \frac{y_{t+1}-\widehat{\mu}_2(t+1)}{h} & \cdots & \frac{y_{t+1}-\widehat{\mu}_2(t+1)}{h-h_1+1} \\
 & \frac{\boldsymbol{d}_{t,h-2}}{h} & \cdots & \frac{\boldsymbol{d}_{t,h-h_1-1}}{h-h_1+1}
\end{pmatrix} \\
&= \left(\frac{\{y_{t+1}-\widehat{\mu}_2(t+1)\}^2}{h+1}+\frac{\boldsymbol{d}_{t,h-1}^{T}\boldsymbol{d}_{t,h-1}}{h+1}, \ldots\right. \\
&\qquad \left., \frac{\{y_{t+1}-\widehat{\mu}_2(t+1)\}\{y_{t-h_1+1}-\widehat{\mu}_2(t-h_1+1)\}}{h-h_1+1}+\frac{\boldsymbol{d}_{t-h_1,h-h_1-1}^{T}\boldsymbol{d}_{t,h-h_1-1}}{h-h_1+1}\right).
\end{aligned}
$$

Making use of $\boldsymbol{d}_{t-d,h-d-1}^T \boldsymbol{d}_{t,h-d-1}/(h-d+1) \sim \boldsymbol{d}_{t-d,h-d}^T \boldsymbol{d}_{t,h-d}/(h-d+2)$ provides (10) for h sufficiently large.

Update of $\widehat{\rho}_t$ in model (18)

Note that

$$\begin{aligned} \left(\sum_{i=0}^{h-1} \widehat{\varepsilon}_{t-i}^2\right)^{-1} &= \left(\widehat{\varepsilon}_t^2 - \widehat{\varepsilon}_{t-h}^2 + \sum_{i=1}^{h} \widehat{\varepsilon}_{t-i}^2\right)^{-1} \\ &\approx \left(\sum_{i=1}^{h} \widehat{\varepsilon}_{t-i}^2\right)^{-1} - \left(\sum_{i=1}^{h} \widehat{\varepsilon}_{t-i}^2\right)^{-2} (\widehat{\varepsilon}_t^2 - \widehat{\varepsilon}_{t-h}^2) \end{aligned}$$

so that the inverse can approximated by recursive updating. Setting $R_{2,t}^{-1} = (\sum_{i=1}^h \widehat{\varepsilon}_{t-i}^2)^{-1}$ one gets $R_{2,t+1}^{-1} \approx R_{2,t}^{-1} - R_{2,t}^{-2}(\widehat{\varepsilon}_t^2 - \widehat{\varepsilon}_{t-h}^2)$. The numerator $R_{1,t} = \sum_{i=1}^h \widehat{\varepsilon}_{t-i}\widehat{\varepsilon}_{t-i+1}$ can be updated by $R_{1,t+1} = R_{1,t} + \widehat{\varepsilon}_t\widehat{\varepsilon}_{t+1} - \widehat{\varepsilon}_{t-h}\widehat{\varepsilon}_{t-h+1}$.

References

Basseville, M. and Nikiforov, I. V. (1993). *Detection of Abrupt Changes, Theory and Application.* Englewood Cliffs, New Jersey, USA: Prentice Hall.

Bauer, M., Gather, U., and Imhoff, M. (1999). Analysis of high dimensional data from intensive care medicine. In R. Payne & P. Green (Eds.), *Proceedings in Computational Statistics*, pp. 185–190. Springer-Verlag, Berlin.

Brockwell, P. J. and Davis, R. A. (1987). *Time Series: Theory and Methods.* Springer-Verlag, Berlin, New York.

Chu, C. K., Glad, I. K., Godtliebsen, F., and Marron, J. (1998). Edge-preserving smoothers for image processing (with discussion). *J. Amer. Statist. Assoc.* **93**, 526–541.

Daumer, M. (1997). Online monitoring of change points. *Biomedizinische Technik* **42**, 95–96.

Daumer, M. (1999). *Verfahren und Vorrichtung zur Erkennung von Driften, Sprüngen und/oder Ausreißern von Meßwerten.* Patent PCT/DE 99/01820.

Daumer, M. and Falk, M. (1998). On-line change-point detection (for state space models) using multi-process kalman filters. *Linear Algebra and its Applications* **284**, 125–135.

Fahrmeir, L. and Künstler, R. (1999). Penalized likelihood smoothing in robust state space models. *Metrika* **49**, 173–191.

Fan, J. and Gijbels, I. (1996). *Local Polynomial Modelling and its Applications.* London: Chapman and Hall.

Gather, U., Bauer, M., Imhoff, M., and Löhlein, D. (1998). Statistical pattern detection in univariate time series of intensive care on-line monitoring data. *Intensive Care Medicine* **24**, 1305–1314.

Hall, P. and Titterington, D. M. (1992). Edge-preserving and peak-preserving smoothing. *Technometrics* **34**, 429–440.

Imhoff, M. and Bauer, M. (1996). Time series analysis in critical care monitoring. *New Horizons* **4**, 519–531.

McDonald, J. A. and Owen, A. B. (1986). Smoothing with split linear fits. *Technometrics* **28**, 195–208.

Müller, H.-G. and Stadtmüller, U. (1999). Discontinuous versus smooth regression. *Ann. Statist.* **27**, 299–337.

Seifert, B. and Gasser, T. (2000). Data adaptive ridging in local polynomial regression. *Journal of Comput. and Graph. Statistics* **9**, 338–360.

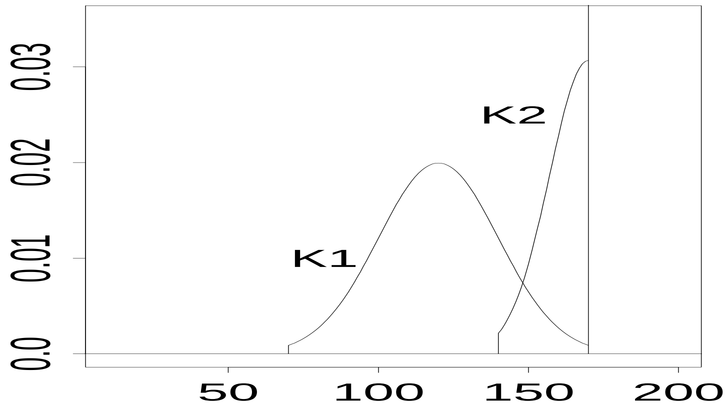

Figure 1: Kernel positions for an estimate at t=170.

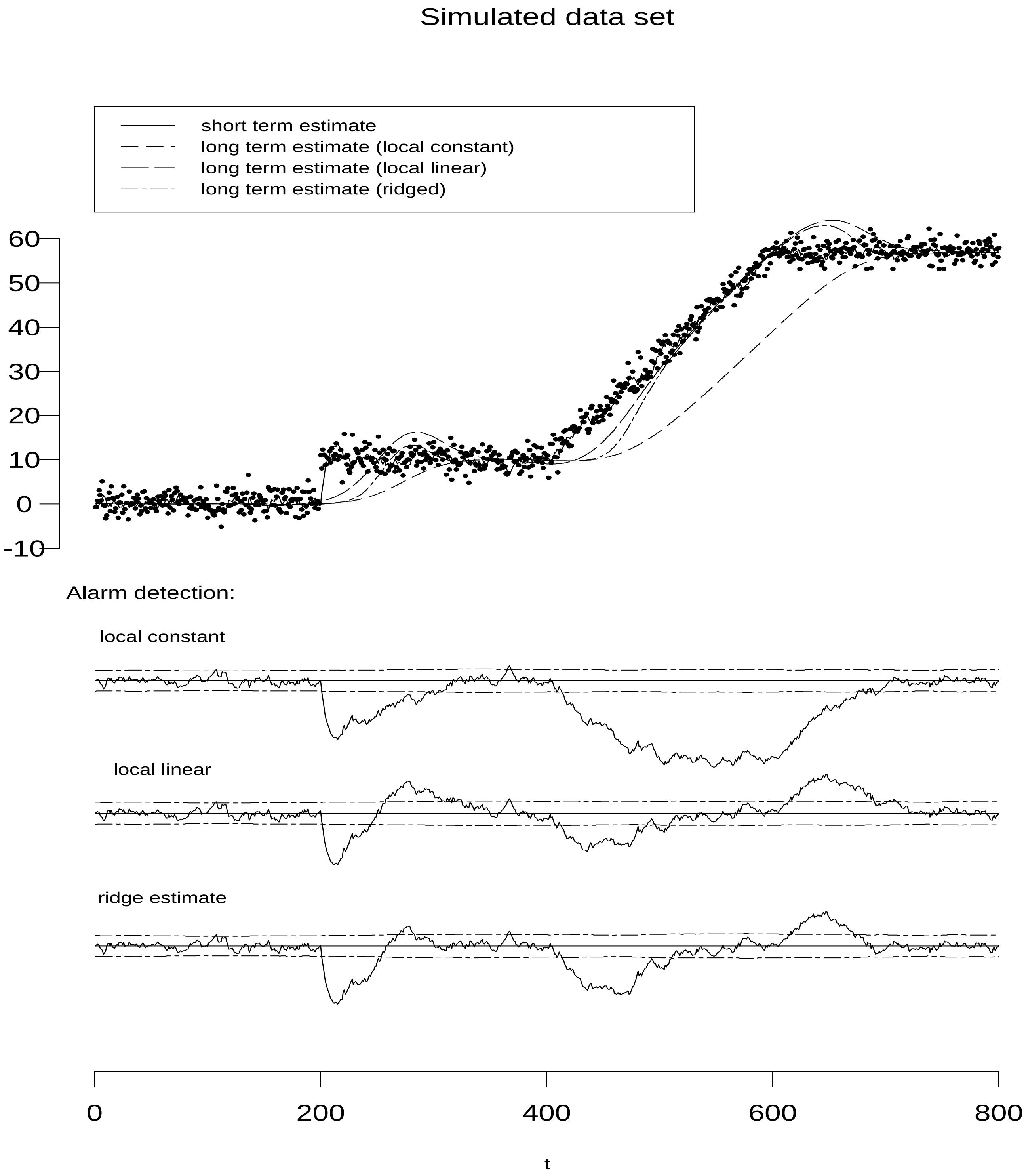

Figure 2: Simulated data set with local constant, local linear and ridged long term estimate using the alarm detection rule (3).

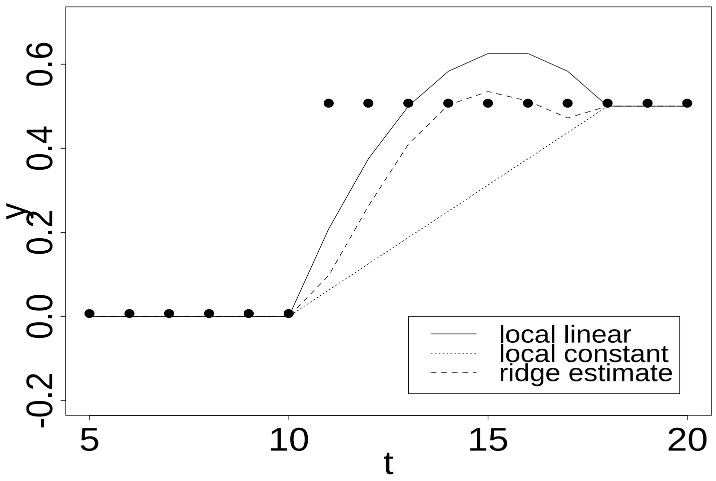

Figure 3: Tutorial on different behavior of local constant, local linear and ridge estimate after a jump.

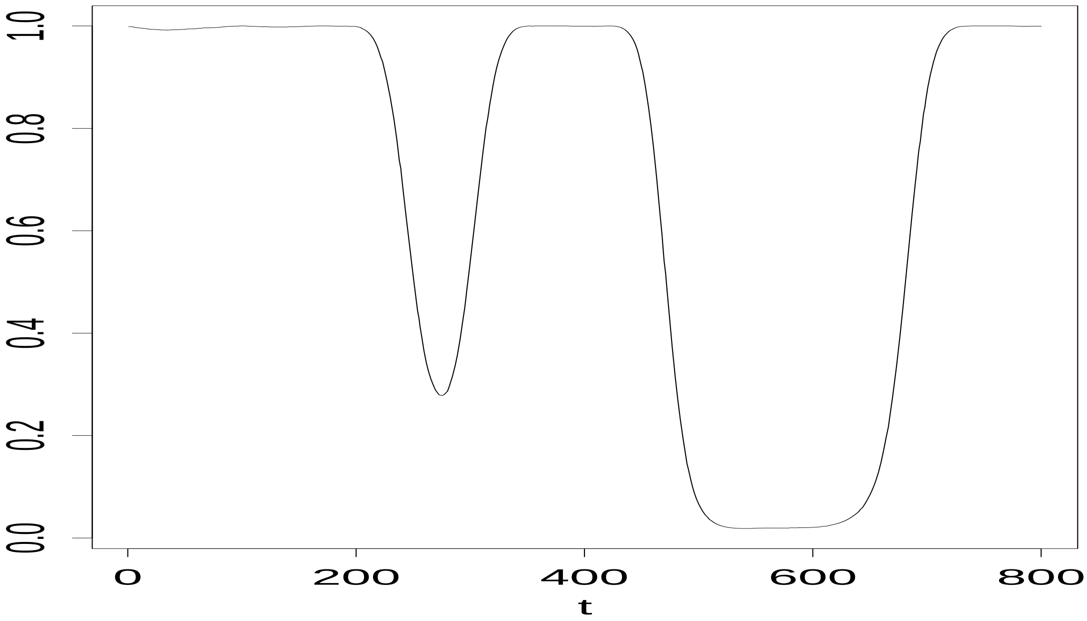

Figure 4: Development of the ridge parameter λ_t over time for the simulated data set analyzed in Fig. 2.

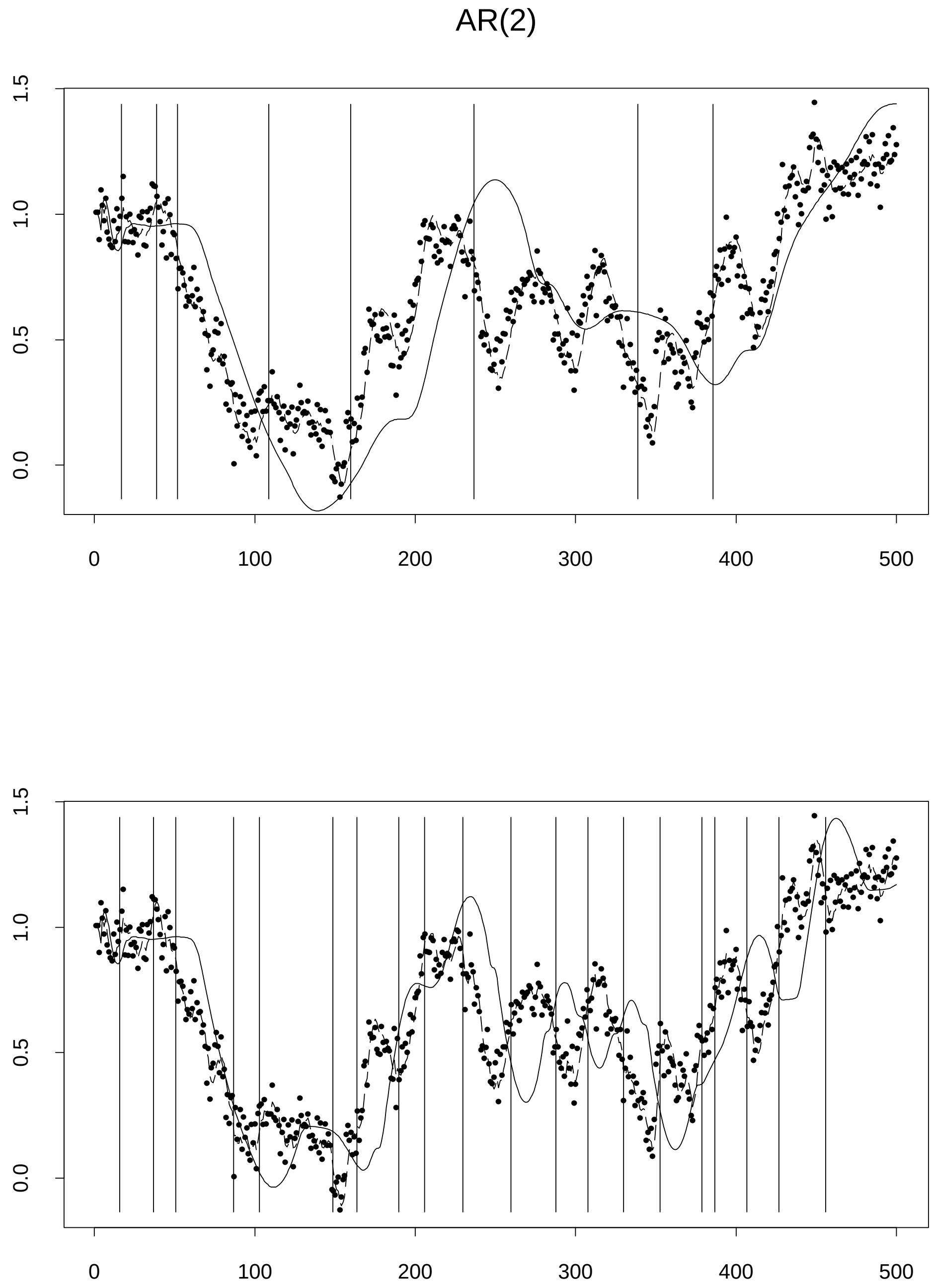

Figure 5: Alarm detection for simulated AR(2) process using $h_1 = 120$ (top) resp. $h_1 = 60$ (bottom). $\widehat{\mu}_1$ and $\widehat{\mu}_2$ are represented by the solid line resp. the dashed line. Vertical lines indicate the detection of breakpoints.

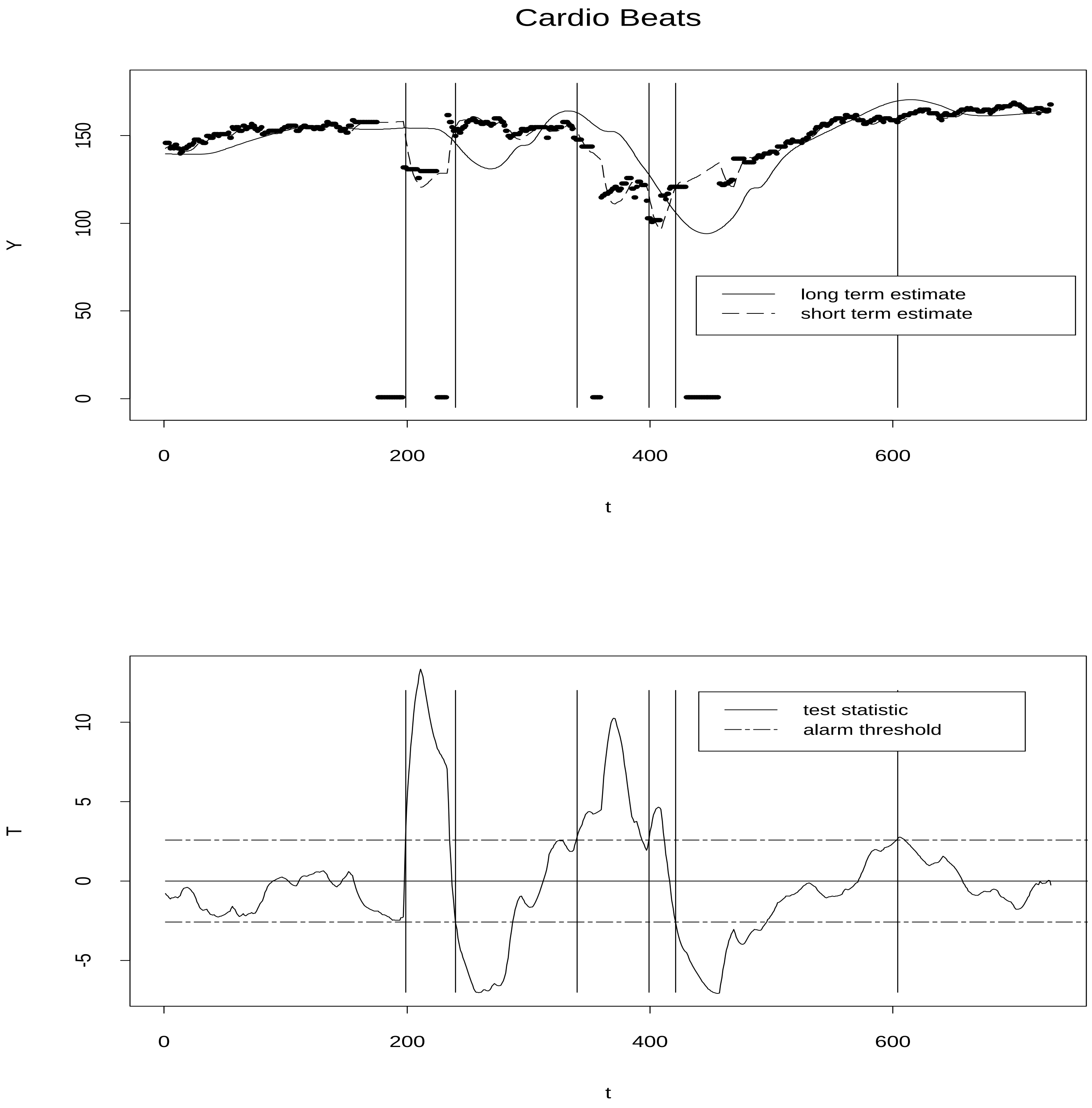

Figure 6: Cardio data with long and short term estimates. In the bottom the test statistic T_t is compared with the quantile $u_{0.995} = 2.58$. Vertical lines indicate the detection of breakpoints.

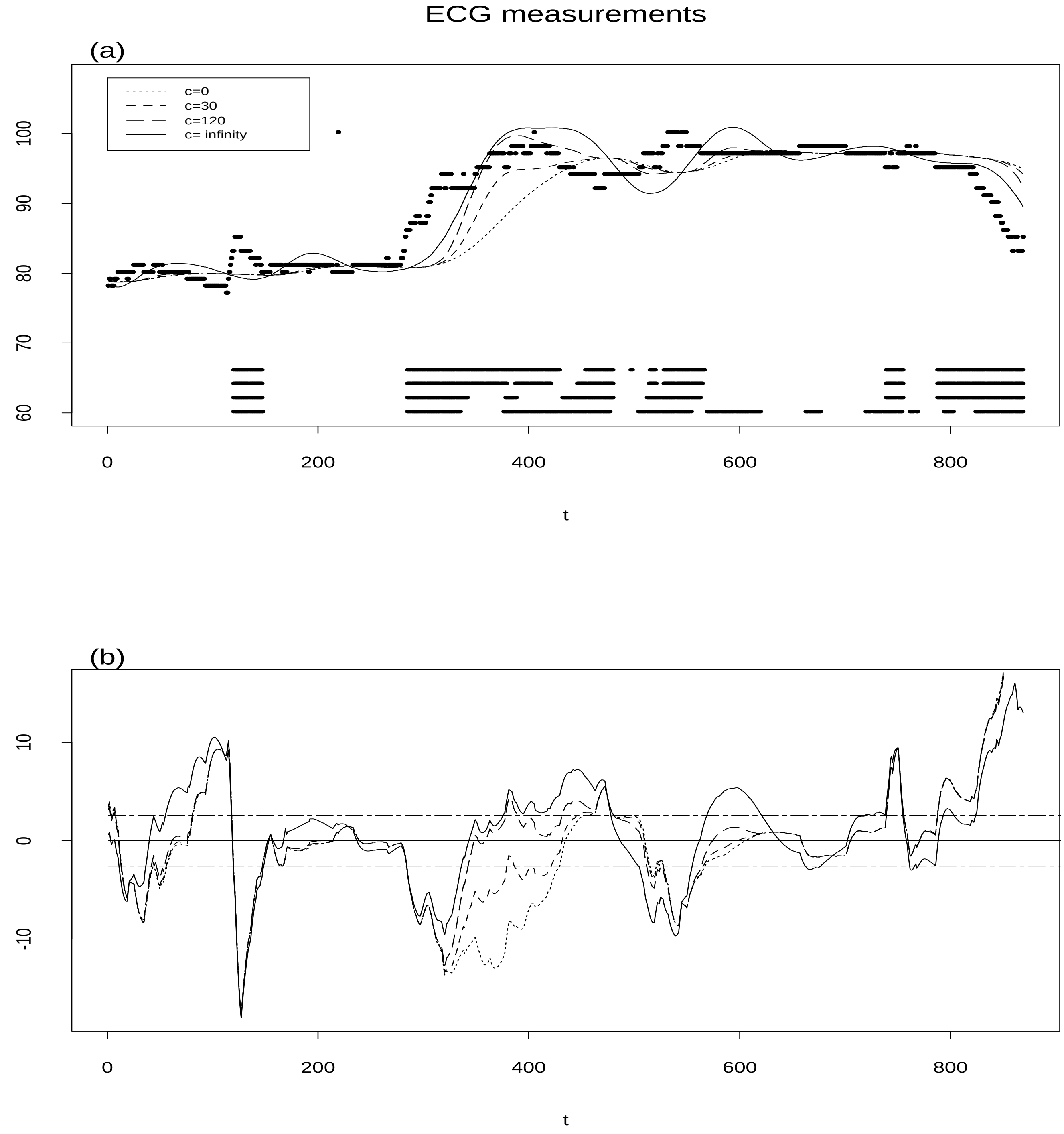

Figure 7: (a) ECG data with long term estimates for different degrees of ridging, using $h_1 = 150, h_2 = 25, h = 200$. The lines in the bottom indicate the alarm periods for $c = 0$ (top), $c = 30, c = 120$, $c = \infty$ (bottom). Alarm signals for $t < 100$ are ignored, since the algorithm needs sufficient data points to work. (b) Test statistic T_t for $c = 0, \ldots, c = \infty$, degrees of ridging symbolized like in (a). Alarm thresholds (horizontal lines) at ± 2.58.

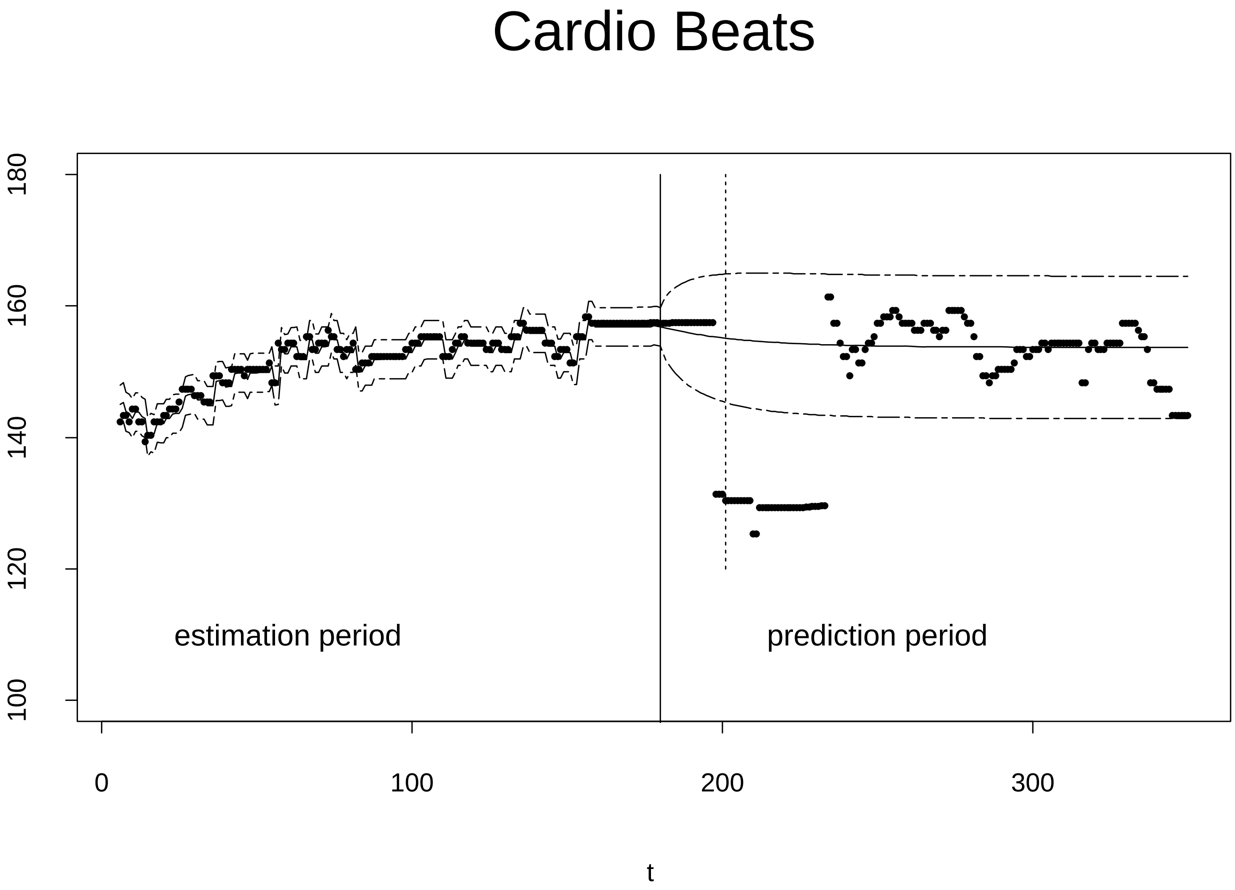

Figure 8: Cardio data with predicted values (solid line), 95% confidence bands (dashed lines) and alarm detection at t=201 (dotted line).

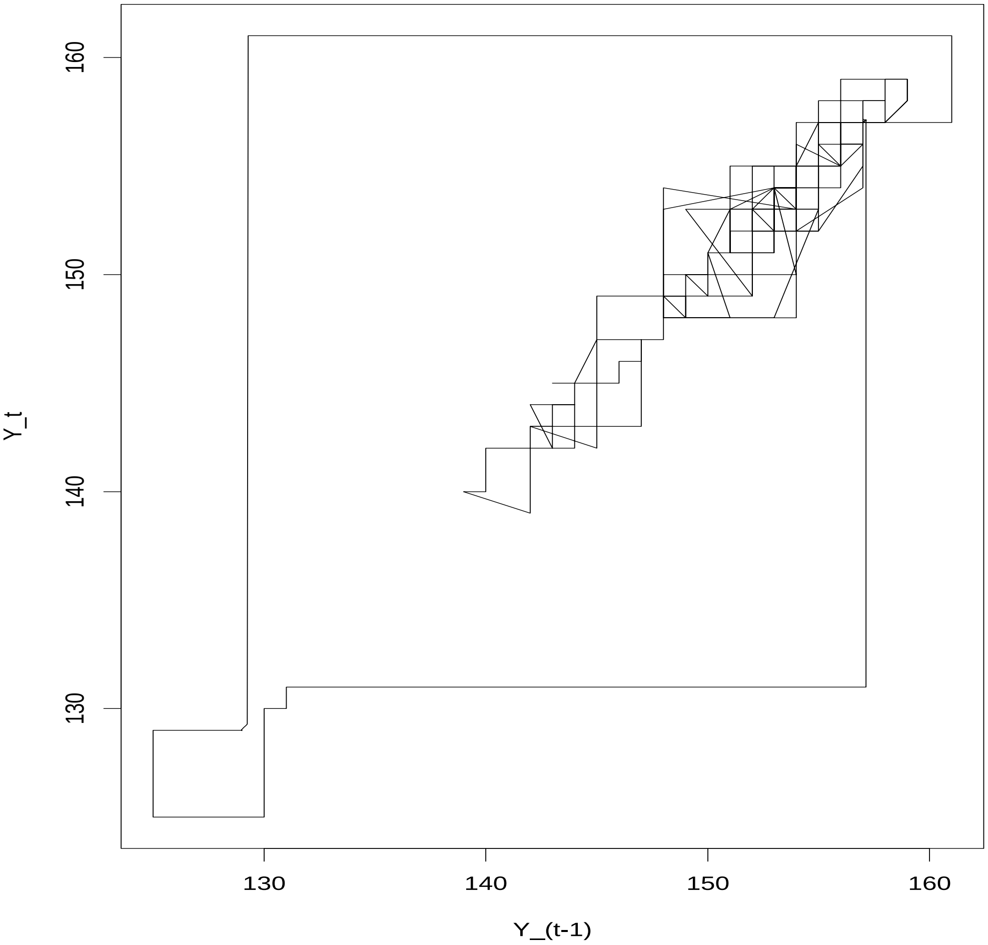

Figure 9: Cardio data from example 4.1. (for $t = 1, \ldots, 380$) plotted in a phase space.

List of Figures