Weierstraß-Institut
für Angewandte Analysis und Stochastik

im Forschungsverbund Berlin e.V.

Preprint ISSN 0946 – 8633

Structural adaptive smoothing by Propagation-Separation-methods

Jörg Polzehl[1] and Vladimir Spokoiny[1]

submitted: 14th November 2005

[1] Weierstrass Institute
for Applied Analysis and Stochastics,
Mohrenstr. 39, 10117
Berlin, Germany
E-Mail: `polzehl@wias-berlin.de`

No. 1068

Berlin 2005

[3] Supported by the DFG Research Center MATHEON "Mathematics for key technologies" in Berlin

2000 *Mathematics Subject Classification.* 62G05.

Key words and phrases. adaptive weights; local structure; propagation; separation; image processing.

Edited by
Weierstraß-Institut für Angewandte Analysis und Stochastik (WIAS)
Mohrenstraße 39
10117 Berlin
Germany

Fax: + 49 30 2044975
E-Mail: preprint@wias-berlin.de
World Wide Web: http://www.wias-berlin.de/

Abstract

Propagation-Separation stands for the main properties of a new class of adaptive smoothing methods. An assumption that a prespecified type of models allows for a good local approximation within homogeneous regions in the design (structural assumption), is utilized to both recover homogeneous regions and to efficiently estimate the regression function. Locality is defined by pairwise weights. Propagation stands for the unrestricted expansion of weights within homogeneous regions. Separations characterizes the restriction of positive weights to homogeneous regions with respect to the specified model. The procedures have remarkable properties like preservation of edges and contrast, and (in some sense) optimal reduction of noise. They are fully adaptive and dimension free. We here provide a short introduction into Propagation-Separation procedures in the context of image processing. Properties are illustrated by a series of examples.

Regression is commonly used to describe and analyze the relation between explanatory input variables X and one or multiple responses Y. In many applications such relations are to complicated to be modeled by a parametric regression function. Classical nonparametric regression, see e.g. Fan and Gijbels (1996); Wand and Jones (1995); Loader (1999); Simonoff (1996) and varying coefficient models, see e.g. Hastie and Tibshirani (1993); Fan and Zhang (1999); Carroll et al. (1998); Cai et al. (2000b), allow for a more flexible form. In this article we describe an approach that allows to efficiently handle discontinuities and spatial inhomogeneity of the regression function in such models.

1 Nonparametric regression

Let us assume that we have a random sample $Z_1, \ldots, Z_n$ of the form $Z_i = (X_i, Y_i)$. Every X_i is a vector of explanatory variables which determines the distribution of an observed response Y_i. Let the X_i's be valued in the finite dimensional Euclidean space $\mathcal{X} = I\!\!R^d$ and the Y_i's belong to $\mathcal{Y} \subseteq I\!\!R^q$. The explanatory variables X_i may e.g. quantify some experimental conditions, coordinates within an image or a time. The response Y_i in these cases identifies the observed outcome of the experiment, the grey value or color at the given location and the value of a time series, respectively.

We assume that the distribution of each Y_i is determined by a finite dimensional parameter $\theta = \theta(X_i)$ which may depend on the value X_i of the explanatory variable.

1.1 Examples

We use the following examples to illustrate the situation.

Example 1.1 [Homoscedastic nonparametric regression model] This model is specified by the regression equation $Y_i = \theta(X_i) + \varepsilon_i$ with a regression function θ and additive i.i.d. Gaussian errors $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$. We will use this model to illustrate the main properties of our algorithms in a univariate ($d = 1$) setting. The model also serves as a reasonable approximation to many imaging problems. Here the explanatory variables X_i define a two ($d = 2$) or three ($d = 3$) dimensional grid with observed grey values Y_i in each grid point.

Example 1.2 [Inhomogeneous Binary Response model] Here Y_i is a Bernoulli random variable with parameter $\theta(X_i)$, that is, $\boldsymbol{P}(Y_i = 1 \mid X_i) = \theta(X_i)$ and $\boldsymbol{P}(Y_i = 0 \mid X_i) = 1 - \theta(X_i)$. This model occurs e.g. in classification. It is also adequate for binary images.

Example 1.3 [Inhomogeneous Poisson model] Every Y_i follows a Poisson distribution with parameter $\theta = \theta(X_i)$, i.e. Y_i attains nonnegative integer values and $\boldsymbol{P}(Y_i = k \mid X_i) = \theta^k(X_i) e^{-\theta(X_i)}/k!$. Such a situation frequently occurs in low intensity imaging, e.g. confocal microscopy and positron emission tomography. It also serves as an approximation of the density model, obtained by a binning procedure.

Example 1.4 [Color images] In color images Y_i denotes a vector of values in a 3 dimensional color space at pixel coordinates X_i. A 4th component may code transparency information. The observed vectors Y_i can often be modeled as multivariate Gaussian, i.e. $Y_i \sim N_3(\theta(X_i), \Sigma)$ with some unknown covariance Σ that may depend on θ. Additionally we will usually observe some spatial correlation.

1.2 Local modeling

We now formally introduce our model. Let $\mathcal{P} = (P_\theta, \theta \in \Theta)$ be a family of probability measures on $\mathcal{Y}$ where Θ is a subset of the real line $\mathbb{R}^1$. We assume that this family is dominated by a measure P and denote $p(y, \theta) = dP_\theta / dP(y)$. We suppose that each Y_i is, conditionally on $X_i = x$, distributed with density $p(\cdot, \theta(x))$. The density is parameterized by some unknown function $\theta(x)$ on $\mathcal{X}$ which we aim to estimate.

A global parametric structure simply means that the parameter θ does not depend on the location, that is, the distribution of every "observation" Y_i coincides with P_θ for some $\theta \in \Theta$ and all i. This assumption reduces the original problem to the classical parametric situation and the well developed parametric theory applies here for estimating the underlying parameter θ. In particular, the maximum likelihood estimate $\widetilde{\theta} = \widetilde{\theta}(Y_1, \ldots, Y_n)$ of θ which is defined by maximization of the log-likelihood

$$L(\theta) = \sum_{i=1}^{n} \log p(Y_i, \theta) \tag{1}$$

is root-n consistent and asymptotically efficient under rather general conditions.

Such a global parametric assumption is typically too restrictive. The classical nonparametric approach is based on the idea of localization: for every point x, the parametric assumption is only fulfilled locally in a vicinity of x. We therefore use a local model concentrated in some neighborhood of the point x.

The most general way to describe a local model is based on weights. Let, for a fixed x, a nonnegative weight $w_i = w_i(x) \le 1$ be assigned to the observations Y_i at X_i, $i = 1, \ldots, n$. When estimating the local parameter $\theta(x)$, every observation Y_i is used with the weight $w_i(x)$. This leads to the local (weighted) maximum likelihood estimate

$$\widetilde{\theta}(x) = \operatorname*{argsup}_{\theta \in \Theta} \sum_{i=1}^{n} w_i(x) \log p(Y_i, \theta). \tag{2}$$

Note that this definition is a special case of a more general local linear (polynomial) likelihood modeling when the underlying function θ is modelled linearly (polynomially) in x, see e.g. Fan et al. (1998). However, our approach focuses on the choice of localizing weights in a data-driven way rather than on the method of local approximation of the function θ.

A common example of choosing the weights $w_i(x)$ is defined by weights of the form $w_i(x) = K_{\mathrm{loc}}(\boldsymbol{l}_i)$ with $\boldsymbol{l}_i = |\rho(x, X_i)/h|^2$ where h is a bandwidth, $\rho(x, X_i)$ is the Euclidean distance between x and the design point X_i and K_{loc} is a *location kernel.* This approach is intrinsically based on the assumption that the function θ is smooth. It leads to a local linear (polynomial) approximation of $\theta(x)$ within a ball of some small radius h centered in the point x, see e.g. Tibshirani and Hastie (1987); Hastie and Tibshirani (1993); Fan et al. (1998); Carroll et al. (1998); Cai et al. (2000a).

An alternative approach is *Localization by a window.* This simply restricts the model to a subset (window) $U = U(x)$ of the design space which depends on x, that is, $w_i(x) = \mathbf{1}(X_i \in U(x))$. Observations Y_i with X_i outside the region $U(x)$ are not used when estimating the value $\theta(x)$. This kind of localization arises e.g. in the regression tree approach, in change point estimation, see e.g. Müller (1992); Spokoiny (1998), and in image denoising, see Qiu (1998); Polzehl and Spokoiny (2003) among many others.

In our procedure we do not assume any special structure for the weights $w_i(x)$, that is, any configuration of weights is allowed. The weights are computed in an iterative way from the data. In what follows we identify the set $W(x) = \{w_1(x), \ldots, w_n(x)\}$ and the local model in x described by these weights and use the notation

$$L(W(x), \theta) = \sum_{i=1}^{n} w_i(x) \log p(Y_i, \theta).$$

Then $\widetilde{\theta}(x) = \operatorname{argsup}_\theta L(W(x), \theta)$. For simplicity we will assume the case where $\theta(x)$ describes the conditional expectation $\boldsymbol{E}(Y|x)$ and the local estimate is obtained explicitly as

$$\widetilde{\theta}(x) = \sum_i w_i(x) Y_i / \sum_i w_i(x). \tag{3}$$

The quality of estimation heavily depends on the localizing scheme we selected. We illustrate this issue by considering kernel weights $w_i(x) = K_{\mathrm{loc}}(|\rho(x, X_i)/h|^2)$ where the kernel K_{loc} is supported on $[0, 1]$. Then the positive weights $w_i(x)$ are concentrated within the ball of radius h at the point x. A small bandwidth h leads to a very strong localization. In particular, if the bandwidth h is smaller than the distance from x to the nearest neighbor, then the resulting estimate coincides with the observation at x. The larger bandwidth we select, the more noise reduction can be achieved. However, the choice of a large bandwidth may lead to the estimation bias, if the local parametric assumption of a homogeneous structure is not fulfilled in the selected neighborhood.

The classical approach to solving this problem is based on a model selection idea. One assumes a set of bandwidth-candidates $\{h_k\}$ to be given, and one of them is selected in a data-driven way to provide the optimal quality of estimation. The global bandwidth selection problem assumes the same kernel structure of localizing schemes $w_i(x)$ for all points x and only one bandwidth h has to be specified. In the local model selection approach, the bandwidth h may vary with the point x. See e.g. Fan et al. (1998) for more details.

We employ a related but more general approach. We consider a family of localizing models, one per design point X_i, and denote them as $W_i = W(X_i) = \{w_{i1}, \ldots, w_{in}\}$. Every W_i is built in an iterative data-driven way and its and support may vary from point to point. The method of constructing such localizing schemes is discussed in the next section.

2 Structural adaptation

Let us assume that for each design point X_i the regression function θ can be well approximated by a constant within a local vicinity $U(X_i)$ containing X_i. This serves as our structural assumption.

Our estimation problem can now be viewed as consisting of two parts. In order to efficiently estimate the function θ in a design point X_i we need to describe a local model, i.e. to assign weights $W(X_i) = \{w_{i1}, \dots, w_{in}\}$. If we knew the neighborhood $U(X_i)$ by an oracle we would define local weights as $w_{ij} = w_j(X_i) = I_{X_j \in U(X_i)}$ and use these weights to estimate $\theta(X_i)$. Since θ and therefore $U(X_i)$ are unknown the assignments will have to depend on the information on θ that we can extract from the observed data. If we have good estimates $\widehat{\theta}_j$ of $\theta(X_j)$ we can use this information to infer on the set $U(X_i)$ by testing the hypothesis

$$H : \theta(X_j) = \theta(X_i). \tag{4}$$

A weight w_{ij} can be assigned based on the value of a test statistic T_{ij}, assigning zero weights if $\widehat{\theta}_j$ and $\widehat{\theta}_i$ are significantly different. This provides us with a set of weights $W(X_i) = \{w_{i1}, \dots, w_{in}\}$ that determines a local model in X_i.

Given the local model we can then estimate our function θ in each design point X_i by (2).

We utilize both steps in an iterative procedure. We start with a very local model in each point X_i given by weights

$$w_{ij}^{(0)} = K_{\mathrm{loc}}(l_{ij}^{(0)}) \quad \text{with} \quad l_{ij}^{(0)} = |X_i - X_j|/h^{(0)} \tag{5}$$

The initial bandwidth $h^{(0)}$ is chosen very small. K_{loc} is a kernel function supported on $[-1, 1]$, i.e. weights vanish outside a ball $U_i^{(0)}$ of radius $h^{(0)}$ centered in X_i. We then iterate two steps, estimation of $\theta(x)$ and refining the local models. In the kth iteration new weights are generated as

$$\begin{aligned} w_{ij}^{(k)} &= K_{\mathrm{loc}}(l_{ij}^{(k)}) K_{\mathrm{st}}(s_{ij}^{(k)}) \quad \text{with} && (6) \\ l_{ij}^{(k)} &= |X_i - X_j|/h^{(k)} \quad \text{and} \quad s_{ij}^{(k)} = T_{ij}^{(k)}/\lambda. && (7) \end{aligned}$$

The bandwidth h is increased by a constant factor with each iteration k. The test statistic for (4)

$$T_{ij}^{(k)} = N_i^{(k)} \mathcal{K}(\widehat{\theta}_i^{(k-1)}, \widehat{\theta}_j^{(k-1)}) \tag{8}$$

with $N_i = \sum_j w_{ij}$ is used to specify the penalty $s_{ij}^{(k)}$. This term effectively measures the statistical difference of the current estimates in X_i and X_j. In (8) the term $\mathcal{K}(\theta, \theta')$ denotes the Kullback-Leibler distance of the probability measures P_θ and $P_{\theta'}$.

Additionally we may introduce a kind of memory in the procedure, that ensures that the quality of estimation will not be lost with iterations. This basically means that we compare a new estimate $\widetilde{\theta}_i^{(k)}$ with the previous estimate $\widehat{\theta}_i^{(k-1)}$ to define a memory parameter $\eta_i = K_m(\boldsymbol{m}_i^{(k)})$ with

$$\boldsymbol{m}_i^{(k)} = \tau^{-1} \sum_j K_{\mathrm{loc}}(\boldsymbol{l}_{ij}^{(k)}) \mathcal{K}(\widetilde{\theta}_i^{(k)}, \widehat{\theta}_i^{(k-1)}). \tag{9}$$

This leads to an estimate

$$\widehat{\theta}_i^{(k)} = \eta_i \widetilde{\theta}^{(k)}(X_i) + (1-\eta_i)\widehat{\theta}_i^{(k-1)}. \tag{10}$$

2.1 Adaptive weights smoothing

We now formally describe the resulting algorithm.

- **Initialization:** Set the initial bandwidth $h^{(0)}$, $k=0$ and compute, for every i the statistics

$$N_i^{(k)} = \sum_j w_{ij}^{(k)}, \quad \text{and} \quad S_i^{(k)} = \sum_j w_{ij}^{(k)} Y_j \tag{11}$$

and the estimates

$$\widehat{\theta}_i^{(k)} = S_i^{(k)} / N_i^{(k)} \tag{12}$$

using $w_{ij}^{(1)} = K_{\mathrm{loc}}(\boldsymbol{l}_{ij}^{(1)})$. Set $k=1$ and $h^{(1)} = c_h^{(0)}$.

- **Adaptation:** For every pair i, j, compute the penalties

$$\boldsymbol{l}_{ij}^{(k)} = |X_i - X_j| / h^{(k)}, \tag{13}$$

$$\boldsymbol{s}_{ij}^{(k)} = \lambda^{-1} T_{ij}^{(k)} = \lambda^{-1} N_i^{(k-1)} \mathcal{K}(\widehat{\theta}_i^{(k-1)}, \widehat{\theta}_j^{(k-1)})). \tag{14}$$

Now compute the weights $w_{ij}^{(k)}$ as

$$w_{ij}^{(k)} = K_{\mathrm{loc}}\big(\boldsymbol{l}_{ij}^{(k)}\big) K_{\mathrm{st}}\big(\boldsymbol{s}_{ij}^{(k)}\big)$$

and specify the local model by $W_i^{(k)} = \{w_{i1}^{(k)}, \ldots, w_{in}^{(k)}\}$.

- **Local estimation:** Now compute new local MLE estimates $\widetilde{\theta}^{(k)}(X_i)$ of $\theta(X_i)$ as

$$\widetilde{\theta}^{(k)}(X_i) = S_i^{(k)} / \widetilde{N}_i^{(k)} \qquad \text{with} \qquad \widetilde{N}_i^{(k)} = \sum_j w_{ij}^{(k)}, \quad S_i^{(k)} = \sum_j w_{ij}^{(k)} Y_l\,.$$

- **Adaptive control:** compute the memory parameter as $\eta_i = K_{\text{me}}(\boldsymbol{m}_i^{(k)}))$. Define

$$\begin{aligned} \widehat{\theta}_{X_i}^{(k)} &= \eta_i \widetilde{\theta}_{X_i}^{(k)} + (1-\eta_i)\widehat{\theta}_{X_i}^{(k-1)} \quad \text{and} \\ N_i^{(k)} &= \eta_i \widetilde{N}_i^{(k)} + (1-\eta_i) N_i^{(k-1)} \end{aligned}$$

- **Stopping:** Stop if $h^{(k)} \geq h_{\max}$, otherwise set $h^{(k)} = c_h h^{(k-1)}$, increase k by 1 and continue with the adaptation step.

2.2 Choice of parameters - Propagation condition

The proposed procedure involves several parameters. The most important one is the scale parameter λ in the statistical penalty s_{ij} . The special case $\lambda = \infty$ simply leads to a kernel estimate with bandwidth $h_{\max}$. We propose to chose λ as the smallest value satisfying a propagation condition. This condition requires that, if the local assumption is valid globally, i.e. $\theta(x) \equiv \theta$ does not depend on x , then with high probability the final estimate for $h_{\max} = \infty$ coincides in every point with the global estimate. More formally we request that in this case for each iteration k

$$\boldsymbol{E}|\widehat{\theta}^{(k)}(X) - \theta| < (1+\alpha)\boldsymbol{E}|\check{\theta}^{(k)}(X) - \theta| \tag{15}$$

for a specified constant $\alpha > 0$. Here

$$\check{\theta}^{(k)}(X_i) = \sum_j K_{\text{loc}}(\boldsymbol{l}_{ij}^{(k)}) Y_j / \sum_j K_{\text{loc}}(\boldsymbol{l}_{ij}^{(k)}) \tag{16}$$

denotes the nonadaptive kernel estimate employing the bandwidth $h^{(k)}$ from step k . The value λ provided by this condition does not depend on the unknown model parameter θ and can therefore be approximately found by simulations. This allows to select default values for λ depending on the specified family of the probability distribution $\mathcal{P} = (P_\theta, \theta \in \Theta)$. Default values for λ in the examples below are selected for a value of $\alpha = 0.1$.

The second parameter of interest is the maximal bandwidth $h_{\max}$ which controls both numerical complexity of the algorithm and smoothness within homogeneous regions.

The scale parameter τ in the memory penalty $\boldsymbol{m}_i$ can also be chosen to meet the propagation condition (15). The special case $\tau = \infty$ turns off the adaptive control step.

Additionally we specify a number of parameters and kernel functions that have less influence on the resulting estimates. As a default the kernel functions are chosen as $K_{\text{loc}}(x) = K_{\text{me}}(x) = (1-x^2)_+$ and $K_{\text{st}}(x) = e^{-x} I_{x<5}$. If the design is on a grid, e.g. for images, the initial bandwidth $h^{(0)}$ is chosen as the distance between neighboring pixel. The bandwidth is increased after each iteration by a default factor $c_h = 1.25^{1/d}$.

3 An illustrative univariate example

We use a simple example to illustrate the behavior of the algorithm. The data in the upper left of Figure 1 follow a univariate regression model

$$Y_i = \theta(X_i) + \varepsilon_i. \tag{17}$$

The unknown parameter, i.e. the regression function, θ is piecewise constant, the errors ε_i are i.i.d. $N(0,1)$ and the observed $X_i = i$ form a univariate grid. In this situation the statistical penalty takes the form

$$\boldsymbol{s}_{ij}^{(k)} = \frac{1}{2\sigma^2\lambda}(\widehat{\theta}_i^{(k-1)} - \widehat{\theta}_j^{(k-1)})^2 \tag{18}$$

where $\sigma^2 = 1$ denotes the variance of the errors. A robust estimate of the variance is obtained from the data using the interquartile range (IQR) as

$$\widehat{\sigma}^2 = (IQR(\{Y_{i+1} - Y_i\}_{i=1,\dots,n-1})/1.908)^2 \tag{19}$$

and used as a plug-in for σ^2. The propagation condition (15) suggests a value of $\lambda = 4.7$. We employ a value of $\tau = \infty$ disabling the adaptive control step.

We have four regions, differing in size and contrast between them, where the function θ is constant. The regression function is displayed as a black line in the upper right of Figure 1.

The lower part of Figure 1 illustrates the evolution of weights w_{ij} with iteration. The horizontal and vertical axis correspond to index i and j, respectively. The upper row provides $K_{\mathrm{loc}}(\boldsymbol{l}_{ij}^{(k)})$ for iterations $k = 0$ ($h = 1$), $k = 7$ ($h = 5$), $k = 13$ ($h = 18$) and $k = 23$ ($h = 169$). The central row shows the corresponding values $K_{\mathrm{st}}(\boldsymbol{s}_{ij}^{(k)})$. The grey scale ranges from black for 0 to white for 1. The weights $w_{ij}^{(k)}$ (lower row) used in the algorithm are the products of both terms.

The left row corresponds to the initialization step. Here the location penalty effectively restricts the local model in X_i to the point X_i itself. If computed the stochastic penalty would contain some weak information about the structure of the regression function. When we reach step $k = 7$ the location penalty allows for positive weights for up to 16 observations, and therefore less variable estimates. At this stage the test (4) shows a significant difference between estimates in points within the third homogeneous interval and estimates in locations outside this interval. This is reflected in the statistical penalty and therefore the weights. In step $k = 13$ also the second interval is clearly identified. The last column, referring to the 23th iteration and a final bandwidth of $h = 169$ shows the final situation

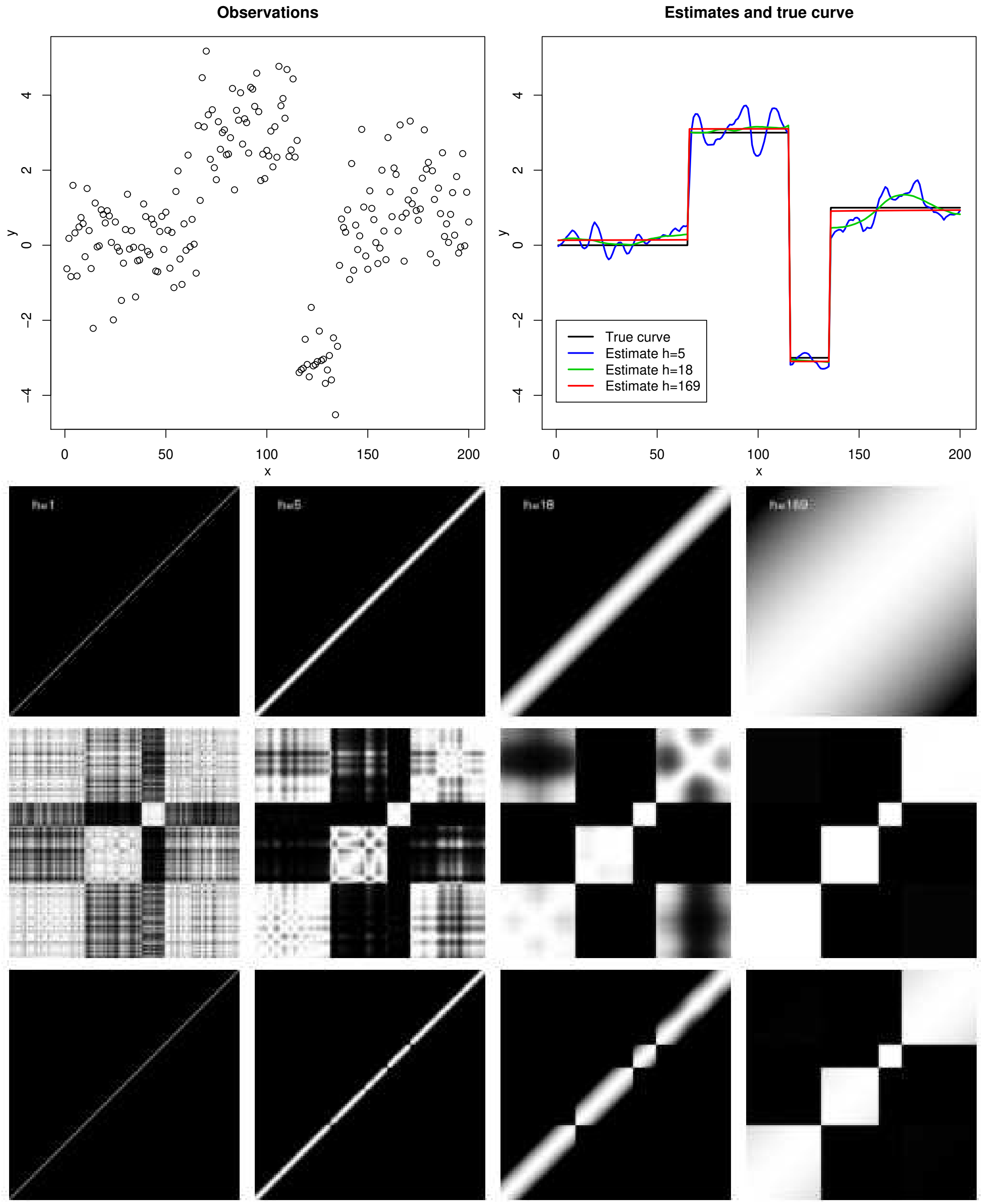

Figure 1: Adaptive weights smoothing for a simple univariate regression problem: data (upper left), regression function (black) and estimates $\widehat{\theta}^{(k)}$ for $k = 7$ ($h = 5$, blue), $k = 13$ ($h = 18$, green) and $k = 23$ ($h = 169$, red) (upper right). The lower part displays the contributions of the location penalty (upper row) and the stochastic penalty (central row) to the weights (lower row) w_{ij} in iteration steps $k = 0, 7, 13$ and 23 (columns).

where the statistical penalty reflects the complete information about the structure and determines the weights. The influence of the location penalty has almost vanished.

What we observe in the iteration process is an unrestricted propagation of weights within homogeneous regions. Two regions with different values of the parameter are separated as values of the statistical penalty s_{ij} increase with decreasing variance of the estimates $\widehat{\theta}_i$ and $\widehat{\theta}_j$ and a large enough contrast $\left|\theta(X_i) - \theta(X_j)\right|^2$. The iteration k where this occurs depends on the size the homogeneous regions, i.e. the potential variance reduction, and the contrast.

The upper right plot in Figure 1 additionally displays the intermediate estimates $\widehat{\theta}^{(k)}, k = 7, 13, 23$ corresponding to the weighting schemes illustrated.

4 Examples and applications

We now provide a series of examples for adaptive weights smoothing in various setups.

4.1 Application 1: Adaptive edge-preserving smoothing in 3D

The algorithm described in subsection 2.1 is essentially dimension free. It can be easily applied to reconstruct 2D and 3D images. We illustrate this using a 3D-MR image of a head. The upper left image in Figure 2 shows the 130th slice of the noisy data cube, consisting of $256 \times 192 \times 256$ voxel. The image is modeled as

$$Y_i = \theta(X_i) + \varepsilon_i, \tag{20}$$

with X_i being coordinates on a 3D-grid and errors ε_i again assumed as i.i.d. Gaussian with unknown variance σ^2. The parameter of interest $\theta(X_i)$ describes a tissue dependend underlying grey value at voxel X_i. Special interest in these images is in identifying tissue borders. Denoising, preferably using an edge-preserving or edge-enhancing filter is a prerequisite step here.

We apply the AWS algorithm from subsection 2.1 using a maximal bandwidth $h_{\max} = 5$. The error variance is estimated from the data. The default value of λ provided by condition (15) for smoothing in 3D with Gaussian errors is $\lambda = 5.9$. The upper right image provides the resulting reconstruction. Note that in the smoothed image the noise is removed while the detailed structure corresponding to tissue borders is preserved. Some deterioration of the image is caused by the structural assumption of a local constant model. This leads to some flattening where $\theta(X_i)$ is smooth.

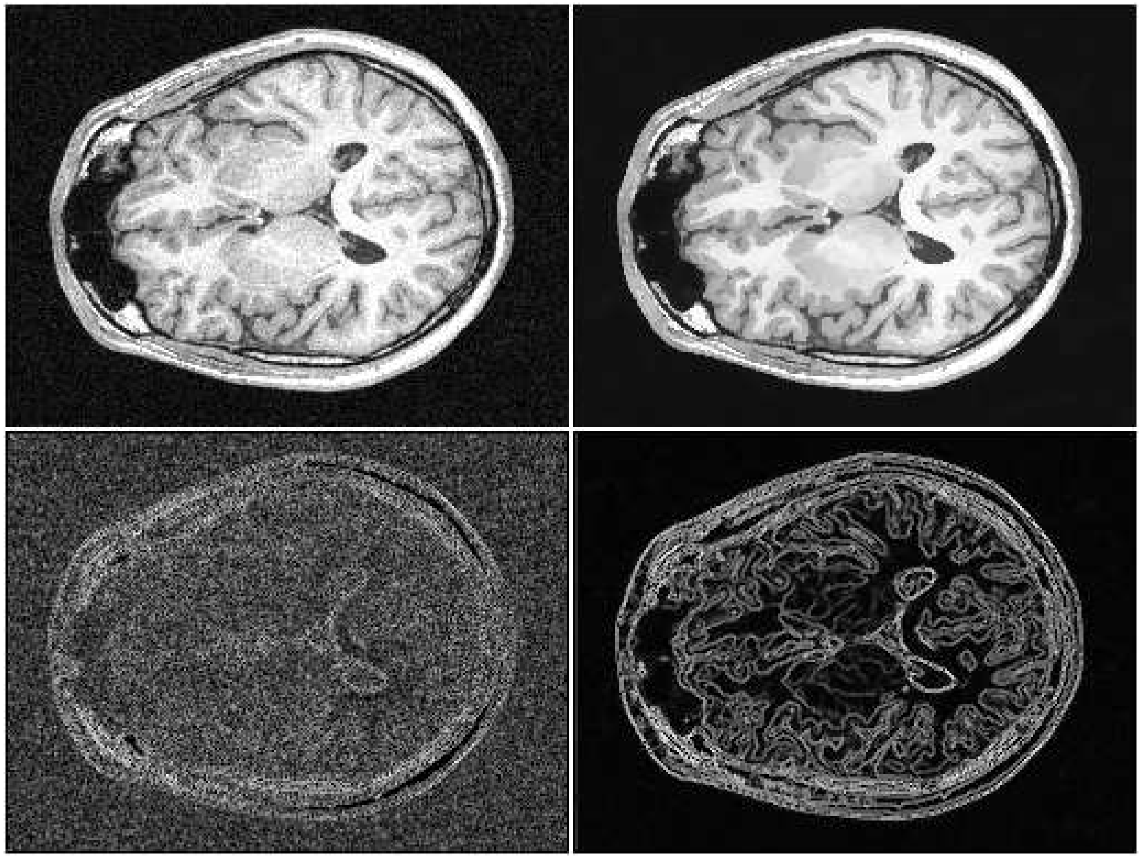

Figure 2: 3D Magnetic resonance imaging (MRI): Slice 130 from a 3D MR image (upper left) and its 3D-reconstruction by AWS (upper right). The lower row shows the result of applying an edge detection filter on both images.

In the bottom row of Figure 2 we provide the absolute value of a Laplacian filter applied to the original noisy image and to the reconstruction obtained by AWS, respectively. We observe an essential enhancement of the tissue borders.

4.2 Examples: Binary and Poisson data

For non-Gaussian data the stochastic penalty s_{ij} takes a different form in (13). The definition is based on the Kullback-Leibler distance $\mathcal{K}$ between the probability measures $P_{\widehat{\theta}_i}$ and $P_{\widehat{\theta}_j}$. For binary data this leads to

$$s_{ij}^{(k)} = \frac{N_i^{(k-1)}}{\lambda}\left(\widehat{\theta}_i^{(k-1)} \log \frac{\widehat{\theta}_i^{(k-1)}}{\widehat{\theta}_j^{(k-1)}} + (1-\widehat{\theta}_i^{(k-1)}) \log \frac{1-\widehat{\theta}_i^{(k-1)}}{1-\widehat{\theta}_j^{(k-1)}}\right) \qquad (21)$$

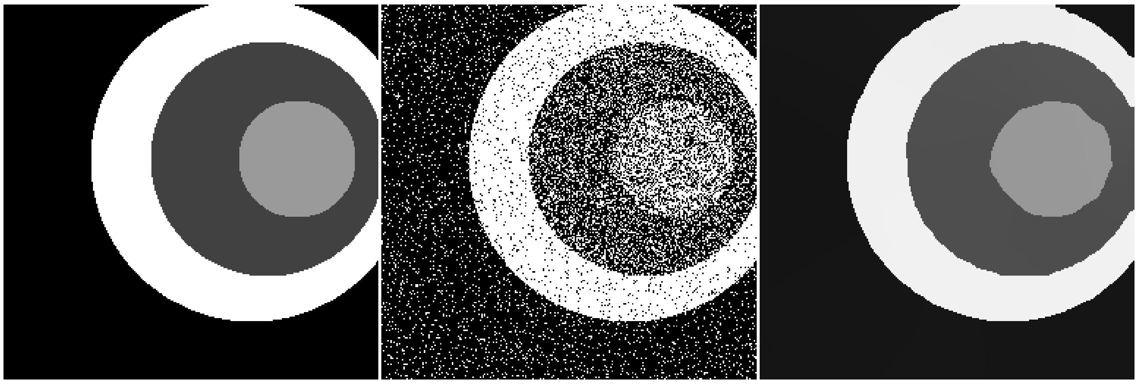

Figure 3: Binary images: Artificial image containing 4 circles with different grey values (left), binary image generated by Bernoulli experiments with pointwise probabilities proportional to the grey values in the left image (center) and reconstructed image of pointwise probabilities (right).

while for Poisson data we get

$$s_{ij}^{(k)} = \frac{N_i^{(k-1)}}{\lambda}\left(\widehat{\theta}_i^{(k-1)} \log \frac{\widehat{\theta}_i^{(k-1)}}{\widehat{\theta}_j^{(k-1)}} - \widehat{\theta}_i^{(k-1)} + \widehat{\theta}_j^{(k-1)}\right). \tag{22}$$

In both cases a special problem occurs. If the estimates $\widehat{\theta}_i$ or $\widehat{\theta}_i$ attain a value at the boundary of the parameter space, i.e. 0 or 1 for binary data or 0 in case of Poisson data, then the Kullback-Leibler-distance between the probability measures $P_{\widehat{\theta}_i}$ and $P_{\widehat{\theta}_j}$ will equal ∞. Such a situation can be avoided by a modification of the algorithm. One solution is to initialize the estimates with the value obtained by the global estimate and to replace the estimate $\widehat{\theta}_j^{(k-1)}$ in (21, 22) by $\widehat{\theta}_{ij}^{(k-1)} = (1-0.5/N_i^{(k-1)})\widehat{\theta}_j^{(k-1)} + 0.5/N_i^{(k-1)}\widehat{\theta}_i^{(k-1)}$ for all following iteration steps.

We use a simple artificial example to demonstrate the performance of the procedure. We start with the image displayed on the left of Figure 3. The image of size 256×256 is composed of 4 regions with distinct grey values. The central image is generated by pixelwise Bernoulli experiments with probabilities $0.08, 0.3, 0.6$ and 0.94, respectively, for the four regions. The image in the right of Figure 3 provides the reconstruction obtained by AWS using a maximal bandwidth $h_{\max} = 100$. The value of λ selected by our propagation condition is $\lambda = 5.9$.

The noisy image in the left of Figure 4 is constructed using the same image structure. Now each grey value is a Poisson count with intensity $0.4, 1.5, 3$ and 4.7, depending the location of the pixel within the image. The right image again provides the reconstruction. A maximal bandwidth $h_{\max} = 50$ and the value $\lambda = 5.4$ provided by the propagation condition (15) for 2D-Poisson images are used.

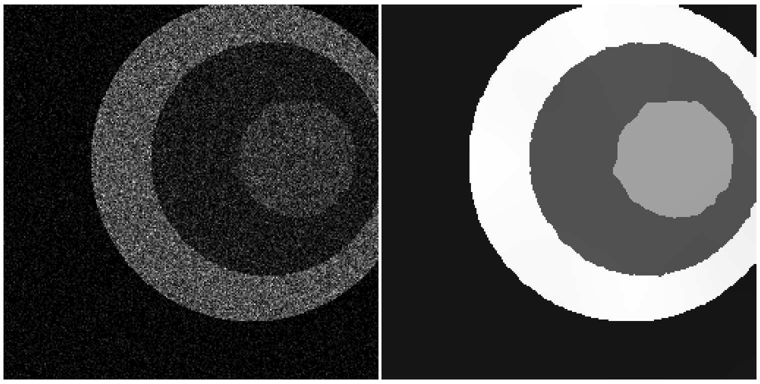

Figure 4: Poisson images: Image generated by Poisson experiments with pointwise intensities proportional to the grey values in the left image of Figure 3 (left) and reconstructed image of pointwise intensities (right).

Note that for both our binary and Poisson image the underlying structure is completely recovered and therefore near optimal estimates of the probabilities and intensities are obtained.

4.3 Example: Denoising of digital color images

In digital color images the information in each pixel consists of a vector of three values. Each value is a intensity in one channel of a three dimensional color space, usually the RGB space. Additionally each pixel may carry some transparency information. Ideally the image is recorded in RAW-format to avoid artifacts caused by lossy image compression and discretization to a low number of color values.

If the image was recorded under bad light conditions, employing a high sensitivity of the sensor, such images can carry a substantial noise. This noise is usually spatially correlated, i.e. colored. Additionally we observe a correlation between the noise components in the three RGB channels.

An appropriate model to describe such a situation is given by

$$Y_{i_h,i_v} = \theta(X_i) + \varepsilon_{i_h,i_v}, \tag{23}$$

where the components of $X_i = (i_h, i_v)$ are the horizontal and vertical image coordinates. Y_{i_h,i_v} , $\theta(X_i)$ and ε_{i_h,i_v} take values in R^3 . The errors follow a distribution with $\boldsymbol{E}\varepsilon_{i_h,i_v} = 0$, $\mathrm{Var}\,\varepsilon_{i_h,i_v} = \Sigma$ and $\boldsymbol{E}\varepsilon^c_{i_h,i_v}\varepsilon^c_{i_h+1,i_v} = \boldsymbol{E}\varepsilon^c_{i_h,i_v}\varepsilon^c_{i_h,i_v+1} = \rho$ for each color channel c. The covariance matrix Σ may vary with the value of θ_{i_h,i_v} .

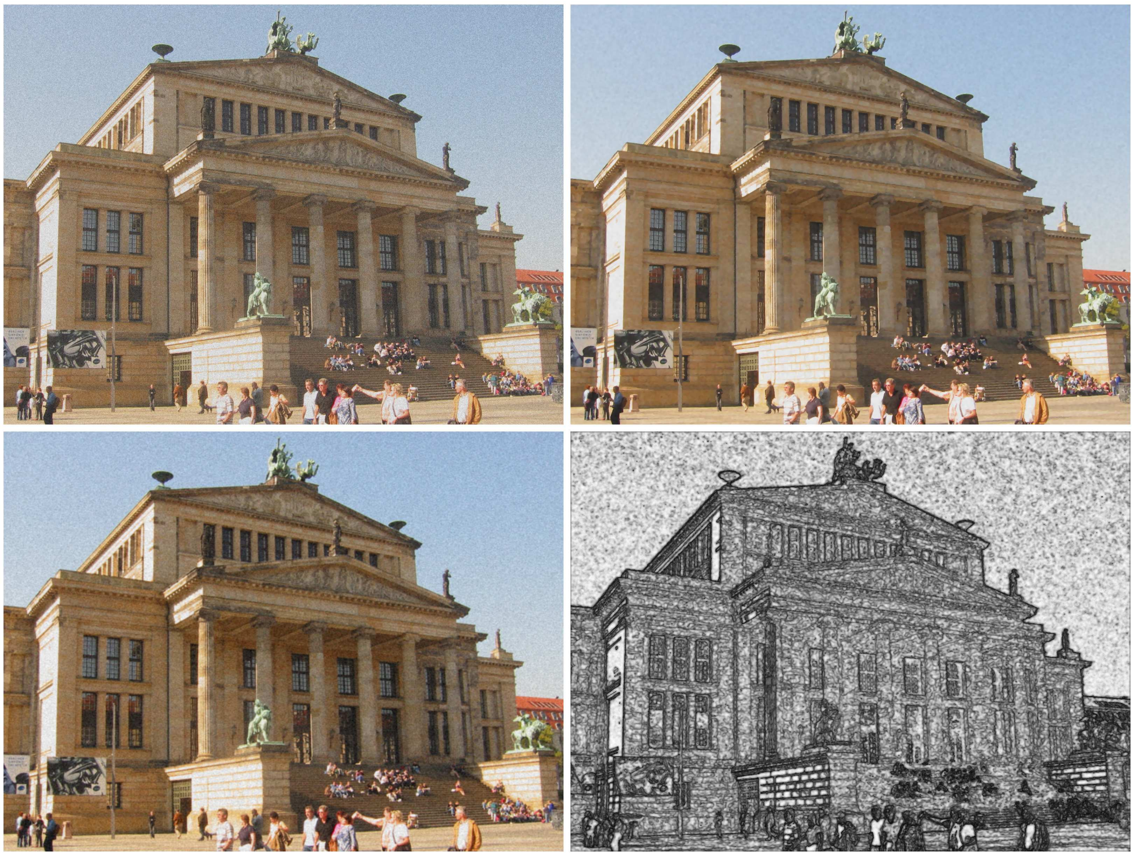

Figure 5: Color images: Image of the Concert Hall at Gendarmenmarkt in Berlin, Germany. The image has been deteriorated by colored noise in all color channels (upper left). Shown are MAE-optimal reconstructions by AWS (upper right) and nonadaptive kernel smoothing (lower left). The lower right image illustrates in each pixel X_i the sum of weights $N_i = \sum_j w_{ij}$ arising for the final estimate.

The algorithm from subsection 2.1 can be applied in this situation with a statistical penalty

$$s_{ij}^{(k)} = \frac{N_i^{(k-1)}}{2\lambda} \big(\widehat{\theta}_i^{(k-1)} - \widehat{\theta}_j^{(k-1)}\big)^\top \varSigma^{-1} \big(\widehat{\theta}_i^{(k-1)} - \widehat{\theta}_j^{(k-1)}\big). \tag{24}$$

The model can often be simplified by transformation of the image to a suitable color space. We observe that a transformation to the YUV or YIQ space uncorrelates the noise between channels, so that a diagonal form of $\varSigma$ seems appropriate under such transformation. In this case error variance can be estimated separately in the three color channels accounting for the spatial correlation.

Figures 5 and 6 provide an example. The upper left image was obtained by deteriorating an digital image showing the Concert Hall at the Gendarmenmarkt in Berlin. The image resolution is 1600×1200 pixel.

Table 1: MAE and MSE in RGB space for the images in Figure 5 and Figure 9.

	Noisy image	AWS reconstruction	Kernel smoothing	local quadratic PS
MAE	3.62 e-2	1.91 e-2	2.25 e-2	1.70 e-2
MSE	2.06 e-3	8.34 e-4	1.12 e-3	6.03 e-4

The original image was transformed from RGB into YUV space. In YUV space the values in the three channels are scaled to fall into the range $(0, 1)$, $(-0.24, 0.19)$ and $(-0.17, 0.46)$, respectively. In each YUV channel colored noise with $\rho = .36$ and standard deviation $\sigma = 0.08, 0.01$ and 0.012, respectively, was added. The resulting noisy image, in RGB space, is shown in the upper left of Figure 5. The upper right image provides the reconstruction by our procedure, using a maximal bandwidth $h_{\max} = 6$. The spatial correlation $\rho = 0.36$ is assumed to be known. The error variance is estimated from the image taking the spatial correlation into account. The statistical penalty selected by the propagation condition (15) for color images with spatially independent noise is $\lambda = 6.90$. This parameter is corrected for the effect of spatial correlation in each iteration.

The lower right image contains in each pixel X_i the value N_i, i.e. the sum of the weights defining the local model in X_i, for the last iteration. We clearly see how the algorithm adapts to the structure in the image, effectively using a large local vicinity of X_i if the pixel belongs to a larger homogeneous region and very small local models if the pixel X_i belongs to a very detailed structure.

Finally we provide the result of the corresponding nonadaptive kernel smoother, i.e. with $\lambda = \infty$, and a bandwidth of $h = 3.1$ a comparison in the lower left of Figure 5. The bandwidth has been chosen to provide a minimal mean absolute error. Table 1 provides the mean absolute error (MAE) and the mean squared error (MSE) for the three images in Figure 5.

Figure 6 provides a detail, with a resolution of 340×545 pixel, from the noisy original (left), the AWS reconstruction (center) and the image obtained by nonadaptive kernel smoothing. The AWS reconstruction produces a much enhanced image at the cost of flattening some smooth areas by its local constant approximation. On the contrary the nonadaptive kernel smoother suffers from a bad compromise between variance reduction and introduction of blurring, or bias, near edges.

Figure 6: Color images: Detail from the images in Figure 5, noisy original (left), AWS reconstruction (center) and kernel smoothing (right).

4.4 Example: Local polynomial Propagation-Separation (PS) approach

Models (17) and (23) assume that the grey or color value is locally constant. This assumption is essentially used in the form of the stochastic penalty s_{ij}. The effect can be viewed as a regularization in the sense that in the limit for $h_{max} \to \infty$ the reconstructed image is forced to a local constant grey value or color structure even if the true image is locally smooth. This is clearly to be seen in the detailed reconstruction in the center of Figure 7 where especially the sculpture looks cartoon-like. Such effects can be avoided if a local polynomial structural assumption is employed. Due to the increased flexibility of such models this comes at the price of a decreased sensitivity to discontinuities.

The Propagation-Separation approach from Polzehl and Spokoiny (2004b) assumes that within a homogeneous region containing $X_i = (i_h, i_v)$, i.e. for $X_j \in U(X_i)$, the grey value or color Y_{j_h,j_v} can be modelled as

$$Y_{j_h,j_v} = \theta(X_i)^\top \Psi(j_h - i_h, j_v - i_v) + \varepsilon_{j_h,j_v}, \tag{25}$$

where the components of $\Psi(\delta_h, \delta_v)$ contain values of basis functions

$$\psi_{m_1,m_2}(\delta_h, \delta_v) = (\delta_h)^{m_1} (\delta_v)^{m_2} \tag{26}$$

for integers $m_1, m_2 \geq 0$, $m_1 + m_2 \leq p$ and some polynomial order p. For a given local

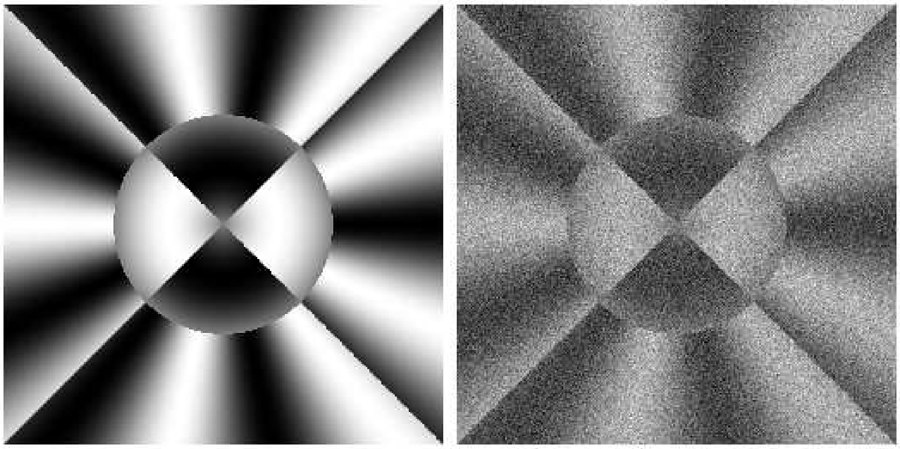

Figure 7: Artificial local smooth image, original (left) and noisy version (right).

model $W(X_i)$ estimates of $\theta(X_i)$ are obtained by local Least Squares as

$$\widetilde{\theta}(X_i) = B_i^{-1} \sum_j w_{ij} \Psi(j_h - i_h, j_v - i_v) Y_{j_h, j_v}, \tag{27}$$

with

$$B_i = \sum_j w_{ij} \Psi(j_h - i_h, j_v - i_v) \Psi(j_h - i_h, j_v - i_v)^\top. \tag{28}$$

The parameters $\theta(X_i)$ are defined with respect to a system of basis functions centered in X_i. Parameter estimates $\widehat{\theta}(X_{j,i})$ in the local model $W(X_j)$ with respect to basis functions centered at X_i can be obtained by a linear transformation from $\widehat{\theta}(X_j)$, see Polzehl and Spokoiny (2004b). In iteration k a statistical penalty can now be defined as

$$s_{ij}^{(k)} = \frac{1}{\lambda 2\sigma^2} \big(\widehat{\theta}^{(k-1)}(X_i) - \widehat{\theta}^{(k-1)}(X_{j,i})\big)^\top B_i \big(\widehat{\theta}^{(k-1)}(X_i) - \widehat{\theta}^{(k-1)}(X_{j,i})\big). \tag{29}$$

In a similar way a memory penalty is introduced as

$$m_{ij}^{(k)} = \frac{1}{\tau 2\sigma^2} \big(\widetilde{\theta}^{(k)}(X_i) - \widehat{\theta}^{(k-1)}(X_i)\big)^\top \widetilde{B}_i^{(k)} \big(\widetilde{\theta}^{(k)}(X_i) - \widehat{\theta}^{(k-1}(X_i)\big) \tag{30}$$

where $\widetilde{B}_i$ is constructed like B_i employing location weights $K_l(l_{ij}^{(k)})$. The main parameters λ and τ are again chosen by a propagation condition requiring free propagation of weights in the specified local polynomial model. A detailed description and discussion of the resulting algorithm and corresponding theoretical results can be found in Polzehl and Spokoiny (2004b).

We use an artificial example to illustrate the behavior of the resulting algorithm. The left image in Figure 7 contains grey values

$$f(x,y) = 0.5\big[1 + \mathrm{sign}(x^2 - y^2)\big\{\sin(7\phi)\mathbf{1}_{\{r \geq 0.5\}} + \sin(\pi r/2)\mathbf{1}_{\{r<0.5\}}\big\}\big]$$

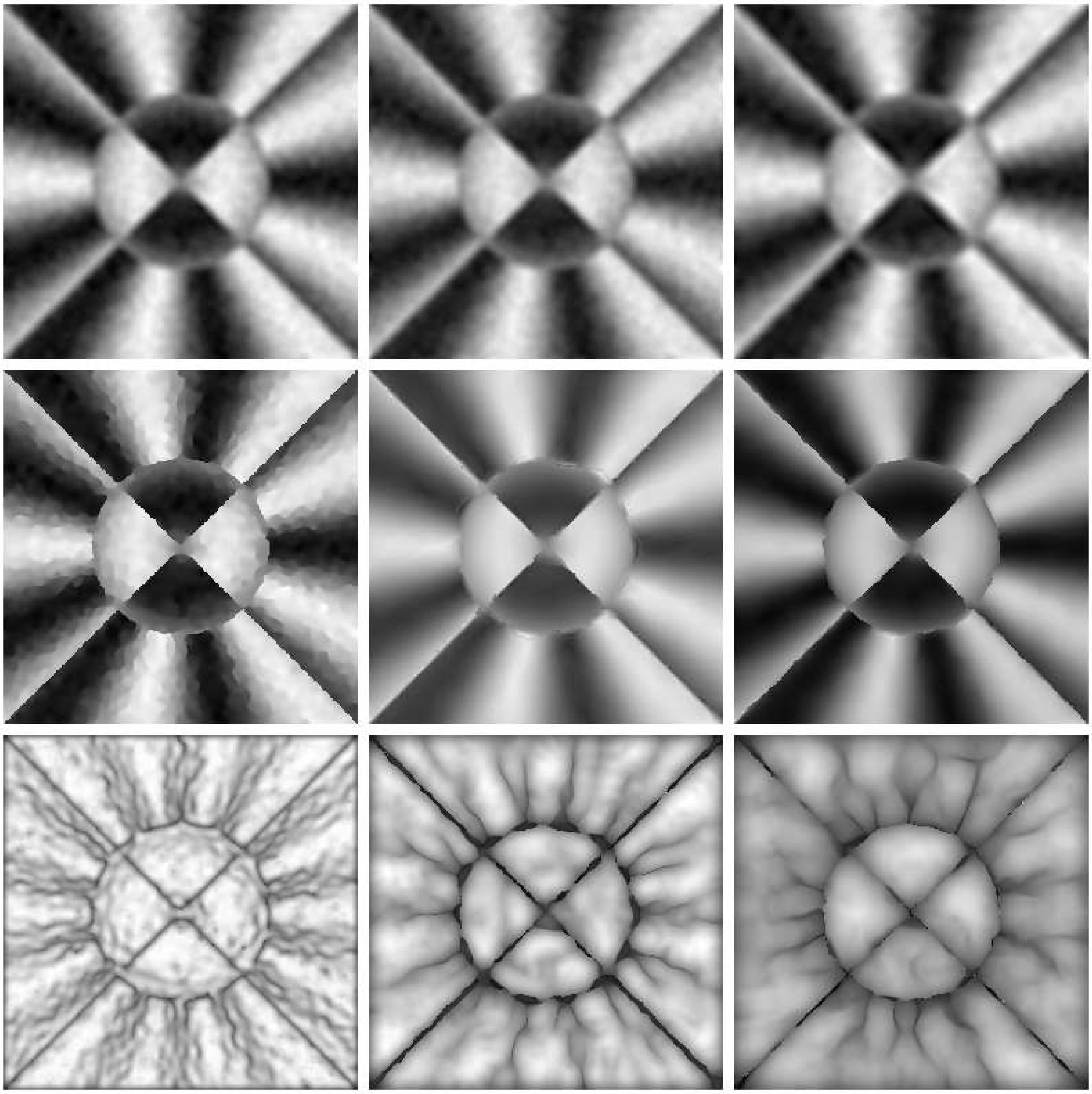

Figure 8: Reconstructions of the noisy image from Figure 7. Upper row: nonadaptive smoothing, central row: structurally adaptive reconstructions, bottom row: pointwise sum of weights used in the structurally adaptive reconstructions. Left column: Local constant smoothing, e.g. kernel smoothing and AWS, central column: Local linear models, right column: local quadratic models. All reconstructions use MAE optimal values for the bandwidth or maximal bandwidth, respectively.

with $x = i/127.5 - 1$, $y = j/127.5 - 1$, $r = \sqrt{x^2 + y^2}$ and $\phi = \arcsin(x/r)$ in locations $i, j = 0, \ldots, 255$. The image is piecewise smooth with sharp discontinuities along diagonals and a discontinuity of varying strength along a circle. The noisy image in the right of Figure 7 contains additive white noise with standard deviation $\sigma = .2$.

Figure 9: Local quadratic reconstruction of the noisy image from Figure 5.

The upper row of Figure 8 provides results obtained by (nonadaptive) kernel, local linear and local quadratic smoothing (from left to right) employing mean absolute error (MAE) optimal bandwidths. The second row gives the reconstructions obtained by the corresponding AWS and Propagation-Separation approaches, again with MAE optimal maximal bandwidths h_{max} . The mean absolute error and mean squared error (MAE) for all six reconstructions together with the employed values of h or h_{max} are contained in Table 2. No adaptive control ($\tau = \infty$) was used for the adaptive procedures. The local constant AWS reconstruction, although clearly improving on all nonadaptive methods, shows clear artifacts resulting from the inappropriate structural assumption used. Also the quality of this result heavily depends on the chosen value of h_{max} . Both local linear and local quadratic PS allow for more flexibility describing smooth changes of grey values. This enables us to use much larger maximal bandwidths, and therefore to obtain more variance reduction without compromising the separation of weights at edges. Best results are obtained by the local quadratic Propagation-Separation algorithm. The bottom row of Figure 8 again illustrates the sum of weights in each pixel generated in the final step of the adaptive procedures.

Figure 10: Upper row: Detail of the original image and same part from its noisy version. Bottom row: Local constant and local quadratic reconstruction.

We now revisit the example from Figure 5. The reconstruction in Figure 9 is obtained by applying the local quadratic Propagation-Separation algorithm with parameters adjusted for the the spatial correlation present in the noisy image. The maximal bandwidth used is $h_{max} = 20$. The statistical penalty selected by the propagation condition for color images with spatially independent noise is $\lambda = 35$. This parameter is again corrected for the effect of spatial correlation in each iteration. Both MAE and MSE of the reconstruction are significantly smaller than for local constant AWS, see Table 1.

The detailed view offered by Figure 10 allows for a more precise judgment on image

Table 2: MAE optimal value of h, MAE and MSE for the images in Figure 8.

	local constant		local linear		local quadratic	
	non-adapt.	AWS (p=0)	non-adapt.	PS (p=1)	non-adapt.	PS (p=2)
h, h_{max}	5.5	6	5.5	15	10	25
MAE	3.27 e-2	3.02 e-2	3.30 e-2	2.10 e-2	3.44 e-2	1.88 e-2
MSE	3.52 e-3	2.17 e-3	3.52 e-3	1.64 e-3	3.52 e-3	1.64 e-3

quality for one of the most structured regions in the image. We provide the same segment of size 300×300 pixel of the original image, its noisy version and both the local constant and local quadratic reconstructions. The local constant reconstruction in general provides more contrast, at the cost of introducing artifacts in smooth regions, see e.g. the sculpture. Local quadratic PS gives a better result with respect to optical impression, MAE and MSE.

5 Concluding remarks

In this article we present a novel adaptive smoothing procedure that has some remarkable properties and a wide potential for applications. We have illustrated with a variety of examples that the approach is essentially dimension free, working in 1D, 2D and even 3D situations. It automatically recovers regions of homogeneity, with respect to a local constant or local polynomial model. As a consequence borders between homogeneous regions are preserved and even enhanced. If the specified local model allows for a good approximation of the unknown function θ this also allows for a significant variance reduction without introduction of bias.

In areas where the function θ is smooth the procedure based on a local constant model is, for large $h_{\max}$, likely to produce a local constant approximation. Nevertheless such a bias introduced at a certain iteration k will be balanced with the variability of the estimates at this iteration. The effect can also be avoided by choosing an appropriate value of $h_{\max}$.

In Polzehl and Spokoiny (2004a) theoretical results are obtained for the case that $\mathcal{P}$ is a one-parameter exponential family. This includes results on propagation, or free extension, of weights within interior sets of homogeneous regions and rates of estimation in regions where the parameter function θ is smooth. Conditions are given for the separation of two homogeneous regions depending on their size and contrast. It is also shown that, up to a constant, at any point the procedure retains the best quality of estimation reached within the iteration process. Related results for the local polynomial Propagation-Separation approach can be found in Polzehl and Spokoiny (2004b).

In the form presented here the procedure is, for dimension $d > 1$, entirely isotropic. It can be significantly improved by introducing anisotropy adaptively, i.e. depending on the information about θ obtained in the iterative process, in the definition of the location penalty.

A reference implementation for the adaptive weights procedure described in subsection 2.1 is available as a package (aws) of the R-Project for Statistical Computing R Development Core Team (2005) from http://www.r-project.org/ .

References

Cai, Z., Fan, J., and Li, R. (2000a). Efficient estimation and inference for varying coefficients models. *J. Amer. Statist. Assoc.*, 95:888–902.

Cai, Z., Fan, J., and Yao, Q. (2000b). Functional-coefficient regression models for nonlinear time series. *J. Amer. Statist. Assoc.*, 95:941–956.

Carroll, R., Ruppert, D., and Welsh, A. (1998). Nonparametric estimation via local estimating equation. *J. Amer. Statist. Assoc.*, 93:214–227.

Fan, J., Farmen, M., and Gijbels, I. (1998). Local maximum likelihood estimation and inference. *J. Roy. Statist. Soc. Ser. B*, 60:591–608.

Fan, J. and Gijbels, I. (1996). *Local polynomial modelling and its applications.* Chapman & Hall, London.

Fan, J. and Zhang, W. (1999). Statistical estimation in varying coefficient models. *Ann. Statist.*, 27:1491–1518.

Hastie, T. and Tibshirani, R. (1993). Varying-coefficient models (with discussion). *J. Roy. Statist. Soc. Ser. B*, 55:757–796.

Loader, C. (1999). *Local Regression and Likelihood.* Springer, New York.

Müller, H. (1992). Change-points in nonparametric regression analysis. *Ann. Statist.*, 20:737–761.

Polzehl, J. and Spokoiny, V. (2000). Adaptive weights smoothing with applications to image restoration. *J. Roy. Statist. Soc. Ser. B*, 62:335–354.

Polzehl, J. and Spokoiny, V. (2001). Functional and dynamic magnetic resonance imaging using vector adaptive weights smoothing. *J. Roy. Statist. Soc. Ser. C*, 50:485–501.

Polzehl, J. and Spokoiny, V. (2003). Image denoising: pointwise adaptive approach. *Ann. Statist.*, 31:30–57.

Polzehl, J. and Spokoiny, V. (2004a). Local likelihood modeling by adaptive weights smoothing. *Probab. Theory Relat. Fields*, in print:.

Polzehl, J. and Spokoiny, V. (2004b). Spatially adaptive regression estimation: Propagation-separation approach. Preprint 998, WIAS.

Qiu, P. (1998). Discontinuous regression surface fitting. *Ann. Statist.*, 26:2218–2245.

R Development Core Team (2005). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.

Simonoff, J. (1996). *Smoothing Methods in Statistics*. Springer, New York.

Spokoiny, V. (1998). Estimation of a function with discontinuities via local polynomial fit with an adaptive window choice. *Ann. Statist.*, 26:1356–1378.

Tibshirani, J. and Hastie, T. (1987). Local likelihood estimation. *J. Amer. Statist. Assoc.*, 82:559–567.

Wand, M. and Jones, M. (1995). *Kernel Smoothing*. Chapman & Hall, London.