Statistical Parametric Maps for Functional MRI Experiments in **R**: The Package fmri

Karsten Tabelow
WIAS Berlin, Matheon

Jörg Polzehl
WIAS Berlin

Abstract

The purpose of the package **fmri** is the analysis of single subject functional magnetic resonance imaging (fMRI) data. It provides fMRI analysis from time series modeling by a linear model to signal detection and publication quality images. Specifically, it implements structural adaptive smoothing methods with signal detection for adaptive noise reduction which avoids blurring of activation areas.

Within this paper we describe the complete pipeline for fMRI analysis using **fmri**. We describe data reading from various medical imaging formats and the linear modeling used to create the statistical parametric maps. We review the rationale behind the structural adaptive smoothing algorithms and explain their usage from the package **fmri**. We demonstrate the results of such analysis using two experimental datasets. Finally, we report on the usage of a graphical user interface for some of the package functions.

Keywords: functional magnetic resonance imaging, structural adaptive smoothing, structural adaptive segmentation, random field theory, multiscale testing, R.

1. Introduction

Neuroscience is an active field that combines challenging scientific questions with interesting methodological developments. It draws expertise from diverse fields as biology, medicine, physics, mathematics, statistics, and computer science. In fact, the emergence of medical imaging techniques has triggered a boost of developments for medical or scientific applications in particular for the examination of the human brain. One of them is functional magnetic resonance imaging (fMRI, Friston *et al.* 2007a; Lazar 2008) which, due to its non-invasive character, has become the most informative tool for *in-vivo* examination of human brain function on small spatial scales. It is utilized both in research as well as clinical applications such as diagnosis and treatment of brain lesions (Vlieger *et al.* 2004; Haberg *et al.* 2004; Henson *et al.* 2005).

As fMRI suffers from a low signal to noise ratio (SNR), noise reduction plays a very important role. Gaussian filtering is applied in standard analysis mostly to increase the SNR and the sensitivity of statistical inference (Friston *et al.* 2007a). At the same time smoothing reduces the number of independent decisions, relaxes the severe multiple test problem and leads to a situation where critical values for signal detection can be assigned using random field theory (RFT, Worsley 1994; Adler 2000). The inherent blurring effect can be ignored as long as the precise shape and extent of the activation area is not crucial. However, experiments are becoming more and more sophisticated and explore columnar functional structures in the brain (Cheng *et al.* 2001) or functions near brain lesions. Here, adaptive noise reduction methods are essential. For a comprehensive introduction into statistical issues in fMRI we refer to Lazar (2008).

In the R package **fmri** (Tabelow and Polzehl 2011) two algorithms from a special class of *structural adaptive smoothing* methods are implemented together with appropriate signal detection: (a) structural adaptive smoothing with RFT for signal detection and (b) structural adaptive segmentation based on multiscale tests. The package **fmri** (version 1.0-0) was first reported in Polzehl and Tabelow (2007b). Since then, the package has evolved significantly and now includes additional features, advances in methodology, and a graphical user interface (GUI). Here, we will report on these updates which refer to the package version 1.4-6. For an introduction to the R environment for statistical computing (R Development Core Team 2010) see Dalgaard (2008) or the materials at `http://www.R-project.org/`. For the use of R in medical imaging problems we refer the reader to the CRAN task view on *Medical Imaging* (Whitcher 2010a).

2. Data handling and linear model

The package **fmri** analyzes fMRI time series using the general linear modeling approach (Friston *et al.* 2007b). The focus of the package is on the use of structural adaptive smoothing methods (Section 3). These methods increase sensitivity of signal detection while limiting the loss in spatial resolution. For an alternative modeling of fMRI data using independent component analysis (ICA) we refer the reader to the package **AnalyzeFMRI** (Marchini and Lafaye de Micheaux 2010; Bordier *et al.* 2011). Other analysis methods for fMRI data are implemented for example in the package **arf3ds4** (activated region fitting, Weeda 2010; Weeda *et al.* 2011) and **cudaBayesreg** (Bayesian multilevel model, Ferreira da Silva 2010, 2011).

2.1. Data formats

fMRI experiments typically acquire time series of full three dimensional brain volumes when a specific stimulus or task is presented or performed, respectively. The package **fmri** provides functionality to read medical imaging data from several formats. It may occasionally be preferable to convert the data using external tools, e.g., **AFNI** (Cox 1996), **MRIcro** (Rorden and Brett 2000), to name only two, or use more specialized R packages, **oro.dicom** (Whitcher 2010b) or **oro.nifti** (Whitcher *et al.* 2010, 2011) for data input. Objects containing fMRI data in the format used within **fmri** may then be easily generated, see documentation and the description below.

The standard format for data recorded by a MR scanner is DICOM (Digital Imaging and

Communications in Medicine; http://medical.nema.org/)[1]. The package **fmri** provides a function for reading DICOM files:

```
R> ds <- read.DICOM(<filename>)
```

where `<filename>` is the name of the file. A data description, without actually reading the data, can be obtained by setting the logical argument `includedata` to `FALSE`. The value returned by the function `read.DICOM` is a list `ds` containing the full header information as a list with the four-byte sequences (group,element) of the DICOM format as elements, the data (if read), and specific header information, e.g., voxel size, image dimension, which are of special interest.

Other formats are more common in the context of fMRI. The ANALYZE format developed at the Mayo Clinic is comprised of two files, the `.img` and `.hdr` files, containing the image data and information about the image acquisition, respectively. Using the **fmri** package the file `<filename>` is read by

```
R> ds <- read.ANALYZE(<filename>)
```

where `<filename>` contains the data as a 4D volume. Data prepared for the analysis software **SPM** (Ashburner *et al.* 2008) generally comes as a series of 3D volume with file names of the form `<prefix+3digits+postfix>`. It is possible to read this series using

```
R> ds <- read.ANALYZE(prefix = <prefix>, numbered = TRUE,
+    postfix = <postfix>, picstart = 1, numbpic = <number>)
```

where `picstart` is the first number in the digits series and `<number>` is the number of volumes.

The ANALYZE format has been further adapted to the NIfTI format by the DFWG (Data Format Working Group, http://nifti.nimh.nih.gov/Members/dfwg/) with improvements in extending the header information and the possibility to merge both files into one (`.nii`). Reading these files is performed by

```
R> ds <- read.NIFTI(<filename>)
```

Note, that currently (version 1.4-6) compressed NIfTI files cannot be handled by **fmri**. Last but not least

```
R> ds <- read.AFNI(<filename>)
```

reads file pairs (HEAD/BRIK) used by *AFNI*.

The input functions for ANALYZE, NIfTI, and AFNI files possess additional arguments `setmask` and `level`. These can be used to classify voxel within and outside the brain based on the mean voxel intensity. If `setmask` is `TRUE` (default) a threshold is set as the `level`-quantile of the mean voxel intensities. Voxel with mean intensity exceeding this threshold are considered to be within the brain. Further computations are restricted to such voxel.

The list object `ds` exploits S3 classes (Chambers 2008) and has class `"fmridata"`. The data cube is stored within the list element `ttt` as a raw vector generated by

---

[1]DICOM is the registered trademark of the National Electrical Manufacturers Association for its standards publications relating to digital communications of medical information.

```
R> writeBin(as.numeric(ttt), raw(), 4)
```

to reduce object size. The data can be extracted from object `ds` in form of a 4D numeric array by

```
R> ttt <- extract.data(ds)
```

The complete header information is contained in the element `header`. Other list elements contain information of special interest (voxel size, data dimension, etc.). If other packages like **oro.nifti** are used for reading the medical imaging data, an object `ds` as described above can be easily created and further processed by **fmri**. Note, that **oro.nifti** provides the function `oro2fmri` for this task.

fMRI datasets can be very large and desktop computer memory may be a limiting resource. Therefore a restriction of the analysis to a special region-of-interest (ROI) may be required and useful. This is provided by the function `cutroi`, where

```
R> dsselected <- cutroi(ds, xind, yind, zind, tind)
```

creates an object containing only a sub-cube of the data defined by the indices `xind`, `yind`, `zind` and `tind`.

The function `summary` provides basic information on objects, for example

```
R> ds <- read.NIFTI("Imagination.nii")
R> summary(ds)
```

generates a short characterization of the second dataset (see Section 4.2):

```
Data Dimension: 64 64 30 105
Data Range    : 0 ... 2934
Voxel Size    : 3.75 3.75 4
File(s) Imagination.nii
```

2.2. Data preprocessing

Data from fMRI experiments acquisition need to be prepared ahead of the quantitative analysis. For example, head motion during the experiment leads to mis-registration of voxels within the data cube at different time points. For group analysis the individual brain has to be mapped (normalized) onto a "standard brain" to identify corresponding brain sections.

There are currently (version 1.4-6) no tools for preprocessing steps like motion-correction, registration and normalization *within* the package **fmri**. There exist many tools to perform these steps in advance, like **SPM**, **AFNI**, **FSL** (Smith *et al.* 2004), or others. Note, that there is an R package **RNiftyReg** to perform image registration (Clayden 2010). fMRI analysis with package **fmri** is currently restricted to single subject analysis with functions to perform group comparisons planned for future versions.

Like all imaging modalities, fMRI suffers from a variety of noise sources, rendering subsequent signal detection difficult. In order to increase sensitivity smoothing of fMRI data is usually part of fMRI preprocessing. It is very important to note, that within the package

fmri smoothing should **not** be applied as preprocessing step. Instead structural adaptive smoothing, which is the main feature of the package, is applied to the statistical parametric map (SPM) derived from the linear model for the time series. If the user intends to use the package **fmri** it is advisable not to smooth the data with third-party tools. Doing so will lead to a loss of essential information and impose a spatial correlation structure, which reduces the effect of structural adaptive smoothing.

2.3. Linear modeling

The observation that voxels with increased neuronal activity are characterized by a higher blood oxygenation level (Ogawa *et al.* 1990, 1992) is known as the BOLD (blood oxygenation level dependent) effect. In fMRI it can be used as a endogenous contrast. Together with the fact that nuclear magnetic resonance is free of high energy radiation, this forms the basis of its appeal as a non-invasive biomarker.

The expected BOLD response corresponding to the stimulus or task of the fMRI experiment may be modeled by a convolution of the experimental design with the hemodynamic response function (HRF). This function characterizes the time delay and modified form of the response in blood oxygenation. Within the package **fmri** the hemodynamic response function $h(t)$ is modeled as the difference of two gamma functions

$$h(t) = \left(\frac{t}{d_1}\right)^{a_1} \exp\left(-\frac{t - d_1}{b_1}\right) - c\left(\frac{t}{d_2}\right)^{a_2} \exp\left(-\frac{t - d_2}{b_2}\right),$$

with default parameters $a_1 = 6$, $a_2 = 12$, $b_1 = 0.9$, $b_2 = 0.9$, and $d_i = a_i b_i$, $i = 1, 2$, $c = 0.35$ and t the time in seconds (Glover 1999). Given the stimulus $s(t)$ as a task indicator function, the expected BOLD response is calculated as the convolution of $s(t)$ and $h(t)$

$$x(t) = \int_0^\infty h(u)s(t - u)du.$$

The resulting function $x(t)$ is evaluated at T scan acquisition times. Such expected BOLD signals may be created by the function

```
R> hrf <- fmri.stimulus(scans = 1, onsets = 1, durations = 1, rt = 3,
+    times = NULL, mean = TRUE, a1 = 6, a2 = 12, b1 = 0.9, b2 = 0.9,
+    cc = 0.35)
```

where `hrf` is a vector of length T, `onsets` is the vector of scan numbers of the stimulus onsets, `durations` is the vector of stimulus durations in number of scans, and `rt` is the time between subsequent volumes. The design can be given in seconds by providing the argument `times`, in this case `durations` should also be given in seconds. One can change the parameters of $h(t)$ to use a slightly different form for the HRF. The vector `hrf` contains the values of the expected BOLD response at the T scan acquisition times. Note, that slice time correction and local modeling of $h(t)$ are currently not available within **fmri**.

Figure 1 shows the vector `hrf` created by `fmri.stimulus()` using

```
R> hrf <- fmri.stimulus(105, c(16, 46, 76), 15, 2)
```
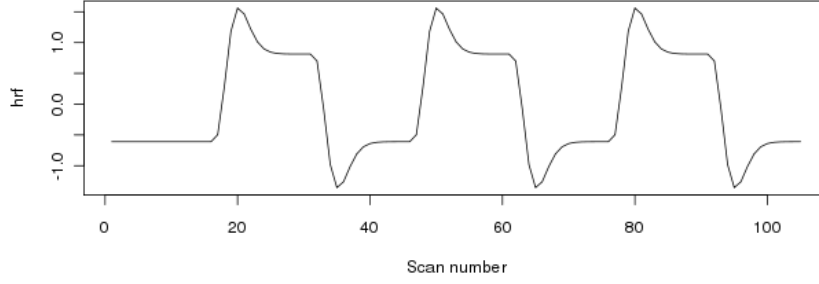
Figure 1: Expected BOLD response for the sports imagination data set.

and used for the analysis of the sports imagination dataset in Section 4.2. Alternatively, a vector of length T (or number of scans) describing an expected BOLD response may be supplied by the user.

For **fmri** we adopt the common view of a linear model for the time series $Y_i = (Y_{it})$ in each voxel i after reconstruction of the raw data and motion correction

$$Y_i = X\beta_i + \varepsilon_i, \quad \mathsf{E}(\varepsilon_i) = 0_T \quad \mathsf{COV}(\varepsilon_i) = \Sigma_i \tag{1}$$

where X denotes the design matrix, containing the expected bold responses `hrf1`, `hrf2`, ..., `hrfq` corresponding to q experimental stimuli as columns (Friston *et al.* 1995; Worsley *et al.* 2002). Additional columns represent a mean effect and a polynomial trend of specified order. Using X we may also model $s \geq 0$ confounding effects `ce` like respiration or heart beat. The columns of X that model the trend are chosen to be orthogonal to the stimulus effects. The components β_{ij} $(j = 1, \ldots, q)$ of β_i describe the effect of stimulus j for voxel i. The additive errors $\epsilon_i = (\epsilon_{i1}, \ldots \epsilon_{iT})$ are assumed to have zero mean and a variance depending on the underlying tissue in voxel i.

The design matrix X may be created using

```
R> x <- fmri.design(cbind(hrf1, ..., hrfq, ce1, ...), order = 2)
```

where `order` is the order of the polynomial trend. Similar to the function `hrf`, x is simply a $p \times T$ matrix ($p = q + s + \text{order}$) and may also be provided directly by the user.

The linear model is evaluated for the fMRI data in `ds` by

```
R> spm <- fmri.lm(ds, x, contrast = 1)
```

In order to access the variability of the parameter estimates $\hat{\beta}_i$ correctly we need to model the structure of Σ_i. In the package **fmri** we assume a temporal auto-correlation following an auto-regressive (AR) model of order 1. The auto-correlation coefficients ρ_i are estimated from the residual vector $r_i = (r_{i1}, \ldots, r_{iT})$ of the fitted model 1 assuming independence, i.e., $\Sigma_i = \sigma_i^2 I_T$. We apply the bias correction given by Worsley *et al.* (2002). Temporal correlation is usually assumed to vary smoothly in space and, by default, we smooth the auto-correlation coefficients using a Gaussian filter in order to stabilize the estimates and increase the number of degrees of freedom (Worsley 2005). Using the resulting estimates $\hat{\rho}_i$ we define a transformation matrix $A_i = \widehat{R}^{-1/2}(\hat{\rho}_i)$, where $\widehat{R}(\rho)$ denotes the correlation matrix under the assumption of an AR(1) process with parameter ρ for voxel i (Tabelow *et al.* 2006).

The model 1 is then transformed into a linear model

$$\widetilde{Y}_{it} = \widetilde{X}_i \beta_i + \tilde{\varepsilon}_{it}, \tag{2}$$

where $\widetilde{Y}_i = A_i Y_i$, $\widetilde{X}_i = A_i X$, and $\tilde{\varepsilon}_i = A_i \varepsilon_i$ with $\mathsf{COV}(\varepsilon_i) \approx \sigma_i^2 I_T$. Least squares estimation of the parameters of interest $\tilde{\beta}_i$ is now straightforward by

$$\tilde{\beta}_i = (\widetilde{X}_i^\top \widetilde{X}_i)^{-1} \widetilde{X}_i^\top \widetilde{Y}_i.$$

The error variance σ_i^2 is estimated from the residuals $\tilde{r}_i$ of the linear model 2 by $\tilde{\sigma}_i^2 = \sum_1^T \tilde{r}_{it}^2/(T-p)$ leading to the voxel-wise estimated covariance matrix

$$\mathsf{COV}(\tilde{\beta}_i) = \tilde{\sigma}_i^2 (\widetilde{X}_i^\top \widetilde{X}_i)^{-1}.$$

Usually special interest is in one of the parameters β_i or a contrast of parameters $\gamma_i = c^\top \beta_i$ represented by the vector c. Hence the result of the parameter estimation in the linear model consists of two three dimensional arrays $\tilde{\Gamma}$ and $\tilde{S}$ containing the estimated effects $\tilde{\gamma}_i = c^\top \tilde{\beta}_i$ and their estimated standard deviations

$$\tilde{s}_i = \sqrt{c^\top \mathsf{COV}(\tilde{\beta}_i) c}.$$

The function `fmri.lm` generates a list object with class attribute `"fmrispm"` containing the arrays $\tilde{\Gamma}$ and $\tilde{S}^2$ as components `beta` and `var`, respectively, and optionally residual information in `"raw"` format, similar to the imaging data. One can use the `extract.data` function with argument `what = "residuals"` to extract the residuals from the object. Note, that changing the contrast of interest c requires re-estimation of the linear model using a new value for the argument `contrast` of `fmri.lm`.

The voxel-wise quotient $\tilde{\theta}_i = \tilde{\gamma}_i / \tilde{s}_i$ of both arrays forms a statistical parametric map $\widetilde{\Theta}$ which is approximately a random t-field (Worsley 1994). Note that the number of degrees of freedom within the t-field depends on the temporal auto-correlation and the smoothing of the AR(1) coefficients (Worsley 2005). All arrays carry a correlation structure induced by the spatial correlation in the fMRI data.

The parameter estimates $\widetilde{\Theta}$ may now be used for a voxel-wise signal detection. A voxel i is classified as activated if the corresponding value $\tilde{\theta}_i$ exceeds a critical value. In this case the estimated BOLD signal (or contrast) significantly deviates from zero. There are typically two intrinsic problems with signal detection. The first is related to the large number of multiple tests in the data cube leading to a large number of false positives if the significance level is not adjusted for multiple testing. Possible solutions are Bonferroni corrections, assuming independence, thresholds obtained from random field theory (Worsley 1994; Adler 2000), depending on sufficient spatial smoothness, or the false discovery rate (FDR, Benjamini and Hochberg 1995; Benjamini and Heller 2007). The second problem is related to small values in $\widetilde{\Theta}$ caused by a large error variance. For both problems smoothing of $\tilde{\Gamma}$ helps, as it reduces the number of independent tests and ideally reduces the variance of the estimated parameters or contrast. In the next section we will review two structural adaptive smoothing methods which are at the core of the package **fmri**.

3. Structural adaptive data processing

3.1. Structural adaptive smoothing

The most common smoothing method for functional MRI data is the Gaussian filter in 3D. It may be easily applied using the fast Fourier transform and guarantees the assumption for RFT which requires a certain smoothness in the data. Unfortunately, non-adaptive smoothing leads to significant blurring and reduces specificity with respect to the spatial extent and form of the activation areas in fMRI. There are several applications where the blurring renders subsequent medical decisions or neuroscientific analysis difficult to interpret; e.g., pre-surgical planning for brain tumor resection (Vlieger *et al.* 2004; Haberg *et al.* 2004; Henson *et al.* 2005) or columnar functional structures (Cheng *et al.* 2001). We have proposed structural adaptive smoothing methods that overcome these drawbacks and allow for increased sensitivity of signal detection while preserving the borders of the activation areas (Tabelow *et al.* 2006; Polzehl *et al.* 2010). Smoothing is generally considered a preprocessing step for the data as is motion-correction etc. However, except for effects from pre-whitening, the order in which non-adaptive spatial smoothing and evaluation of the linear model are performed is arbitrary. Moreover, if the temporal correlations are spatially homogeneous the temporal modeling and spatial smoothing may be interchanged.

Structural adaptive smoothing uses local homogeneity tests for the adaptation (Polzehl and Spokoiny 2006) that are inefficient in the four-dimensional data space. Structural adaptive smoothing is therefore based on the estimates of $\tilde{\Gamma}$ and $\tilde{S}$ obtained by temporal modeling using `fmri.lm`. Parameter estimation in the linear model serves as a variance and dimension reduction step prior to spatial smoothing and therefore allows for a much better adaptation (Tabelow *et al.* 2006).

Structural adaptive smoothing requires an assumption on the spatial homogeneity structure of the true parameter γ_i. We assume a local constant function for γ_i. The null hypothesis for the statistical test of activation is $H_0 : \gamma_i = 0$, i.e., we therefore assume that non-activated areas are characterized by a parameter value of zero. In areas which are activated during the acquisition the parameter values differ from zero and are similar, provided that the BOLD %-changes are similar.

Based on this assumption and the arrays $\tilde{\Gamma}$ and $\tilde{S}$, we use an iterative smoothing algorithm for the SPM that is based on pairwise tests of homogeneity. We specify kernel functions $K_l(x) = (1 - x^2)_+$ and $K_s(x) = \min(1, 2(1 - x)_+)$. For K_l this choice is motivated by the kernels near efficiency in non-adaptive smoothing. Its compact support and simplicity reduces the computational effort. The second kernel should exhibit a plateau near zero, be compactly supported and monotone non-increasing on the positive axis. We denote by $d(i, j)$ the distance between voxel i and j. Let $\mathcal{H} = \{h_k\}_{k=1}^{k^*}$, with $h_0 = 1$, be a series of bandwidths such that

$$\left(\sum_j K_l \left(d(i,j) h_k^{-1} \right) \right)^2 \Big/ \sum_j K_l^2 \left(d(i,j) h_k^{-1} \right)$$

forms a geometric sequence with factor $c_h = 1.25$. This provides, from step to step and in case of non-adaptive smoothing, a variance reduction by factor c_h. At iteration step k we

define for each voxel i and all voxels j within a distance $d(i, j) < h_k$ weights

$$w_{ij}^{(k)} = K_l \left(\frac{d(i, j)}{h_k} \right) K_s \left(s_{ij}^{(k-1)} \right),$$

$$s_{ij}^{(k-1)} = \frac{N_i^{(k-1)}}{\lambda C(k, \rho_s) \tilde{s}_i^2} \left(\tilde{\gamma}_j^{(k-1)} - \tilde{\gamma}_i^{(k-1)} \right)^2. \tag{3}$$

We call $s_{ij}^{(k-1)}$ a statistical penalty as it evaluates the statistical difference between the estimated parameters $\tilde{\gamma}_i^{(k-1)}$ and $\tilde{\gamma}_j^{(k-1)}$ in voxels i and j. Using the weights $w_{ij}^{(k)}$ we compute smoothed estimates of the contrasts

$$\tilde{\gamma}_i^{(k)} = \frac{1}{N_i^{(k)}} \sum_j w_{ij}^{(k)} \tilde{\gamma}_j, \qquad N_i^{(k)} = \sum_j w_{ij}^{(k)}.$$

The parameter λ in (3) can be determined by simulation depending on the error model (but not the actual data) using a propagation condition (Polzehl and Spokoiny 2006).

The term $C(k, \rho_s)$ in (3) provides an adjustment for the effect of spatial correlation in the original data, characterized by a vector of first-order correlation coefficients ρ_s, at iteration k. Optionally, specifying `adaptation = "fullaws"` for the function `fmri.smooth`, causes the term $C(k, \rho_s) \tilde{s}_i^2 / N_i^{(k-1)}$ to be replaced by an estimated variance $\hat{\sigma}_i^{2;(k-1)}$ obtained from the smoothed residuals $\tilde{r}_{it}^{(k-1)} = \sum_j w_{ij}^{(k-1)} \tilde{r}_{jt} / N_i^{(k-1)}$. This leads to additional computational cost but usually improves the results. Finally, the variances of the smoothed estimates (contrasts) are estimated from smoothed residuals using the weighting scheme $w_{ij}^{(k^*)}$ from the final iteration k^*.

Structural adaptive smoothing provides an intrinsic stopping criterion, since the quality of estimates is preserved at later steps of the algorithm. It leads to estimates $\tilde{\Gamma}^{(k^*)}$ and $\tilde{S}^{2;(k^*)}$ and a smoothed SPM $\tilde{\Theta}^{(k^*)}$. In these statistics shape and borders of the activation structure are preserved. As a consequence, compared to non-adaptive smoothing methods, the procedure reduces noise while preserving the resolution of the scan as required by many modern applications (Tabelow *et al.* 2009).

The number of iteration steps k^* determines a maximum achievable variance reduction or equivalently a maximal achievable smoothness. It can be specified by selecting the maximum bandwidth h_{max} in the series h_k as the expected diameter of the largest area of activation. Over-smoothing structural borders is avoided in the algorithm as long as differences between the parameter values are statistically significant at any scale, i.e., for any bandwidth h_k, visited within the iterations. The largest homogeneous region is expected to be the non-activation area, where parameter values do not significantly differ from zero. Within this area, under the hypothesis of no activation, structural adaptive smoothing due to the propagation condition behaves like its non-adaptive counterpart with $\lambda = \infty$.

Structural adaptive smoothing within the package **fmri** is performed by

```
> spm.smooth <- fmri.smooth(spm, hmax = 4, adaptation = "aws")
```

The choice of the adaptation method is specified using the argument `adaptation`. Specifically `adaptation = "none"` uses non-adaptive smoothing, `adaptation = "aws"` or `"fullaws"` corresponds to adaptive smoothing (with improved variance estimates for the latter at the

cost of computation time), and `adaptation = "segment"` uses the structural adaptive seg-mentation described in the next section. Note, `hmax` is the only other parameter for the algorithm that should be specified by the user. Additional parameters of the function have only minor influence on the results and should be considered for experts usage and testing purpose only.

From the final estimates for $k = k^\star$ the random t-field $\tilde{\Theta}^{(k^\star)} = \widetilde{\Gamma}^{(k^\star)}(\tilde{S}^{2;(k^\star)})^{-1/2}$ can be constructed. Under the null hypotheses of no activation the propagation condition ensures a smoothness corresponding to the application of a non-adaptive filter with the bandwidth $h_{max} = h_{k^\star}$. In order to define thresholds for signal detection on such a smooth t-field RFT is applied in the package **fmri** (Adler 2000; Tabelow *et al.* 2006).

Signal detection is performed by function `fmri.pvalue` applied to an object of class `"fmrispm"`.

```
R> pvalue <- fmri.pvalue(spm.smooth)
```

creates a list object of class `"fmripvalue"` containing the p values at each voxel in the list element `pvalues`. For the determination of the p values we do not include effects from the non-stationarity of the random field (Hayasaka *et al.* 2004). This could be done using the objects `spm` and `spm.smooth` but is currently not implemented in `fmri.pvalue`. If no smoothing has been performed then signal detection by FDR may be specified using the argument `mode = "FDR"` in function `fmri.pvalue`. The result `pvalue` is displayed via

```
R> plot(pvalue, anatomic = anatomic, maxpvalue = 0.05)
```

with a possible anatomic image `anatomic` used as an underlay. The anatomic image must be supplied in NIfTI format using `read.NIFTI` or as an array of the same size as the functional data. The significance level for the multiple test corrected p values can be chosen by an appropriate value for `maxpvalue`.

3.2. Structural adaptive segmentation

Structural adaptive smoothing and subsequent signal detection by RFT forms a sequential procedure that depends on the similar behavior of the adaptive method and non-adaptive smoothing under the null hypothesis of no activation. It is desirable to combine adaptive smoothing and signal detection in a way that solves the multiple-comparison problem. Such integration is possible since the information used to generate weighting schemes in the struc-tural adaptive procedure (Tabelow *et al.* 2006) can also be used for signal detection. The re-sulting method is called structural adaptive segmentation (Polzehl *et al.* 2010). This method leads to similar signal detection results, but is conceptually more coherent and requires fewer assumptions concerning the signal detection. For example, it automatically accounts for the spatial smoothness in the data and does not rely on the assumption of stationary errors. It also provides a more computationally efficient algorithm.

Let V be a region-of-interest (ROI), and construct a hypothesis test

$$H : \max_{i \in V} \gamma_i \le \delta \qquad (\text{or} \ \max_{i \in V} |\gamma_i| \le \delta), \tag{4}$$

where δ corresponds to a suitable minimal signal size for γ. The probability to reject the hypothesis in *any* voxel $i \in V$ should be less or equal a prescribed significance level α.

Using results from extreme value theory (Resnick 1987; Polzehl *et al.* 2010) and ideas from multiscale testing (Dümbgen and Spokoiny 2001) a suitable test statistics for this hypotheses is

$$T(\tilde{\Gamma}^{\mathcal{H}}) = \max_{h \in \mathcal{H}} \max_{i \in V} \frac{\left(\tilde{\gamma}_i^{(h)} - \delta\right)}{a_{n(h)}(\nu)\hat{s}_i^{(h)}} - \frac{b_{n(h)}(\nu)}{a_{n(h)}(\nu)},$$

where the degrees of freedom ν of the t statistics $\tilde{\theta}_i$ is, in the non-adaptive case, the shape parameter of the limiting Fréchet distribution (Resnick 1987). The normalizing sequences $a_{n(h)}(\nu)$ and $b_{n(h)}(\nu)$ are selected to achieve, under the hypothesis H, $\delta = 0$ and $\lambda = \infty$, a good approximation of the distribution of $T(\hat{\Gamma}^{\mathcal{H}})$ by the Fréchet distribution Φ_ν.

The adaptive segmentation algorithm employs in each iteration step a test statistic

$$T(\hat{\Gamma})^{(k)} = \max_{i \in V} \frac{\left(\hat{\gamma}_i^{(h)} - \delta\right)}{a_{n_i^{(k)}(h)}(\nu)\hat{s}_i^{(h)}} - \frac{b_{n_i^{(k)}(h)}(\nu)}{a_{n_i^{(k)}(h)}(\nu)}, \tag{5}$$

where $n_i(h) \leq n(h)$ reflects the effect of adaptive weights, to decide whether or not to reject the null hypothesis at iteration k. Adaptive smoothing is performed as long as the hypothesis is not rejected. Otherwise non-adaptive smoothing is restricted to the set of detected deviance's from the hypothesis in either a positive or negative direction. Critical values are determined by simulation under the null hypothesis as quantiles of the distribution of

$$\tilde{T}(\hat{\Gamma}^{\mathcal{H}}) = \max_k T(\hat{\Gamma})^{(k)},$$

for a wide range of n, ν and suitable values of λ. A minimum value of λ has again been selected to fulfill the propagation condition under the null hypothesis (Polzehl *et al.* 2010).

Structural adaptive segmentation combines the structural adaptive smoothing algorithm described in the preceding section with a test using $T(\hat{\Gamma})^{(k)}$ at iteration k. The result provides three segments, one, where the hypotheses of no activation could not be rejected, and two segments for the two-sided alternatives. Note, no p values are determined but the method uses a specified significance level α for determining the critical values for the test. There are several parameters for the structural adaptive segmentation (Polzehl *et al.* 2010). Most, like the choice of kernel function or the rate of increase for the series of bandwidths for the iteration, have only minor influence. These are pre-defined in the package and cannot be changed by the user. The parameter λ, which controls the degree of adaptation, is chosen for any specified error distribution by simulation using the propagation-separation condition (Polzehl *et al.* 2010), and cannot be changed by the user.

In the **fmri** package structural adaptive segmentation may be performed on an object `fmrispm` from the linear model `fmri.lm` by choosing `adaptation = "segment"` in

```
R> spm.segment <- fmri.smooth(spm, hmax = 4, adaptation = "segment",
+     alpha = 0.05)
```

The (multiple test) significance level `alpha` may be chosen within the range 0.01 to 0.2. Additionally, an argument `delta` may be provided for the minimum signal size used in the test statistics (5). The result of the structural adaptive segmentation contains information on three segments that can be visualized using

```
R> plot(spm.segment)
```

Within activated areas the size of the estimated signal is provided as additional color-coded information.

4. Examples

4.1. Auditory experiment (dataset 1)

In our first example we use an auditory dataset recorded by Geraint Rees under the direction of Karl Friston and the Functional Imaging Laboratory (FIL) methods group which is available from the **SPM** website (Ashburner *et al.* 2008). 96 acquisitions were made with scan to scan repeat time RT=7s, in blocks of 6, resulting in 16 blocks of 42s duration. The condition for successive blocks alternated between rest and auditory stimulation, starting with a resting block. Auditory stimulation was with bi-syllabic words presented binaurally at a rate of 60 per minute. The functional data starts at acquisition 4, image fM00223_004. Echo planar images (EPI) were acquired on a modified 2T Siemens MAGNETOM Vision system. Each acquisition consisted of 64 contiguous slices with matrix size 64×64 and 27mm^3 isotropic voxel (http://www.fil.ion.ucl.ac.uk/spm/data/auditory/).

The following script (spm.R) can be used to process the first dataset. As an alternative the package provides a graphical user interface (GUI), see Section 5, to guide the user through the analysis.

```
R> library("fmri")
R> ds <- read.ANALYZE("fM00223/fM00223_", TRUE, "", 4, 96)
R> anatomic <- extract.data(ds)[ , , , 1]
R> hrf <- fmri.stimulus(96, (0:7) * 12 + 7, 6, 7)
R> x <- fmri.design(hrf)
R> spm <- fmri.lm(ds, x)
R> spm.seg <- fmri.smooth(spm, hmax = 4, adaptation = "segment")
R> plot(spm.seg, anatomic)
```

Images can be directly exported from the GUI invoked by plot (Section 5).

4.2. Sports imagination experiment (dataset 2)

For the second example a sports imagination task fMRI scan was performed by one healthy adult female subject within a research protocol approved by the institutional review board of Weill Cornell Medical College. For functional MRI, a GE-EPI sequence with TE/TR = 40/2000 ms was used and 30 axial slices of 4 mm thickness were acquired on a 3T GE system. A field-of-view (FOV) of 24 cm with a matrix size 64×64, yielding in-plane voxel dimensions of 3.75 mm, respectively, was used. The excitation flip angle was 80 degrees. Task and rest blocks had a duration of 30 s and were played out in the following order: rest, task, rest, task, rest, task, rest, totaling 105 repetitions. Before the first block, 6 dummy scans where acquired to allow for saturation equilibrium. The task consisted of imagination of playing tennis.
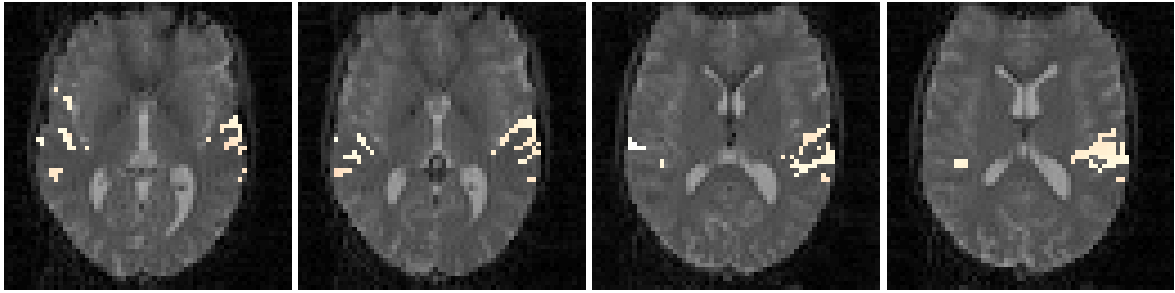
Figure 2: Result of signal detection using structural adaptive segmentation and the dataset 1 (auditory experiment, slices 33, 34, 36, and 37). The color scheme codes the size of the estimated signal in a voxel where a activated segment has been detected by the algorithm. The underlay is the first volume of the time series.

For the anatomy, a sagittal 3D MP-RAGE scan was acquired (matrix 256 x 160 x 110, re-sampled to 256 x 256 x 110, 24 cm FOV, 1.5 mm slice thickness, TR = 8.6 ms, TE = 1.772 ms, delay time = 725 ms, flip angle = 7 degrees).

The following script (`imagination.R`) can be used to process the second dataset.

```
R> library("fmri")
R> library("adimpro")
R> counter <- c(paste("0", 1:9, sep = ""), paste(10:99, sep = ""))
R> ds <- read.NIFTI("Imagination.nii")
R> scans <- 105
R> onsets <- c(16, 46, 76)
R> duration <- 15
R> tr <- 2
R> hrf <- fmri.stimulus(scans, onsets, duration, tr)
R> x <- fmri.design(hrf)
R> spm <- fmri.lm(ds, x)
R> spm.segment <- fmri.smooth(spm, hmax = 4, adaptation = "segment")
R> ds.ana <- read.NIFTI("MPRAGEco.nii")
R> for (slice in 1:30) {
+    img <- plot(spm.segment, ds.ana, slice = slice)
+    write.image(make.image(img, gammatype = "ITU"), file =
+      paste("result", counter[slice], ".png", sep = ""))
+ }
```

4.3. Results

The results of the script `spm.R` run on the first dataset can be seen in Figure 2. Some slices (33, 34, 36, 37) with detected auditory activation have been chosen in the GUI invoked by the `plot` function in the script. The data has been smoothed using the structural adaptive segmentation algorithm with a maximum bandwidth of four voxels and a significance level of 0.05 (default). Figure 2 shows that there does not occur an apparent blurring effect as it would after applying a Gaussian filter with corresponding bandwidth. The color in the activation area codes the size of the estimated BOLD signal.
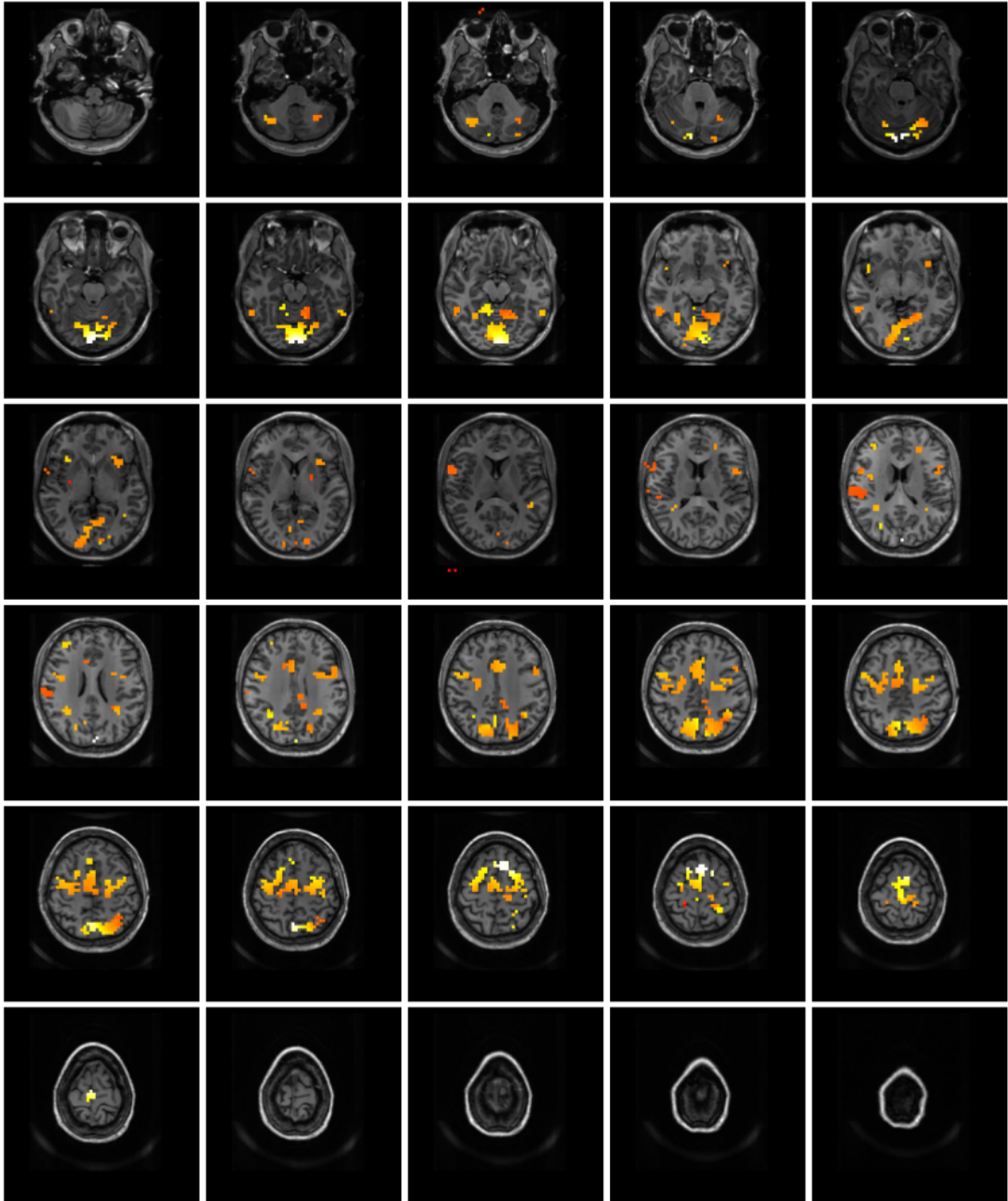
Figure 3: Result of signal detection using structural adaptive segmentation and the second dataset described above. The color scheme codes the size of the estimated signal in a voxel where an activated segment has been detected by the algorithm. The underlay is a sagittal 3D MP-RAGE scan.

Images obtained using the script `imagination.R` and the second dataset are shown in Figure 3. The figure shows all slices of the dataset with detected signals using structural adaptive segmentation with a maximum bandwidth of four voxels and a significance level of 0.05. Activations are found primarily in the supplementary motor area and the premotor areas. The structural adaptive smoothing leads to increased sensitivity without blurring the borders of the activation areas. The color again codes the size of the estimated BOLD signal after adaptive smoothing in arbitrary units using the full range of the color from red to white for the range of estimated signals. Note, that the MP-RAGE scan is not mapped to the EPI data. Therefore good matching between functional and anatomical data is limited to the upper brain slices with the motor activation, which is the main result of this experiment. Some spurious activation outside the brain is due to a pronounced EPI ghost signal as a result of the smoothing algorithm.

5. Organizing the computational work flow (GUI)

In order to provide a user friendly environment we created a graphical user interface (GUI) which guides the user through the work flow described above. The GUI does not provide the full functionality of the package but gives quick results for standard analysis scenarios.

After invoking the package **fmri** the GUI can be started

```
R> library("fmri")
R> fmrigui()
```

and provides step by step instructions. At each step of the analysis help information is available. The following design information needs to be provided to perform the analysis for the auditory experiment.

```
interscan intervals:                       7
scans per session:                        96
Time unit (design) scans or seconds?    scans
number of conditions:                      1
condition name:                      Auditory
onset times:         7 19 31 43 55 67 79 91
duration:                                  6
```

The information may be saved in a text file and re-used. Data may be accessed in AFNI, AN-ALYZE, or NIfTI files. In this example, use the file select box to navigate to the directory containing the ANALYZE files and select the first file within the directory, file `fM00223_004.hdr` in folder `fM00223`. The data are read and basic consistency checks against the experimental design are performed.

In the next step a threshold for mean image intensity is proposed. This threshold is used to define a mask of voxel with mean intensity larger than the threshold. The mask should contain voxel within the brain so that computations are restricted to voxel within this mask. The button `View Mask` provides images of the mask defined by the specified threshold together with density plots of image intensities for centered data cubes of varying sizes. This information is provided to assist the selection of an appropriate threshold.
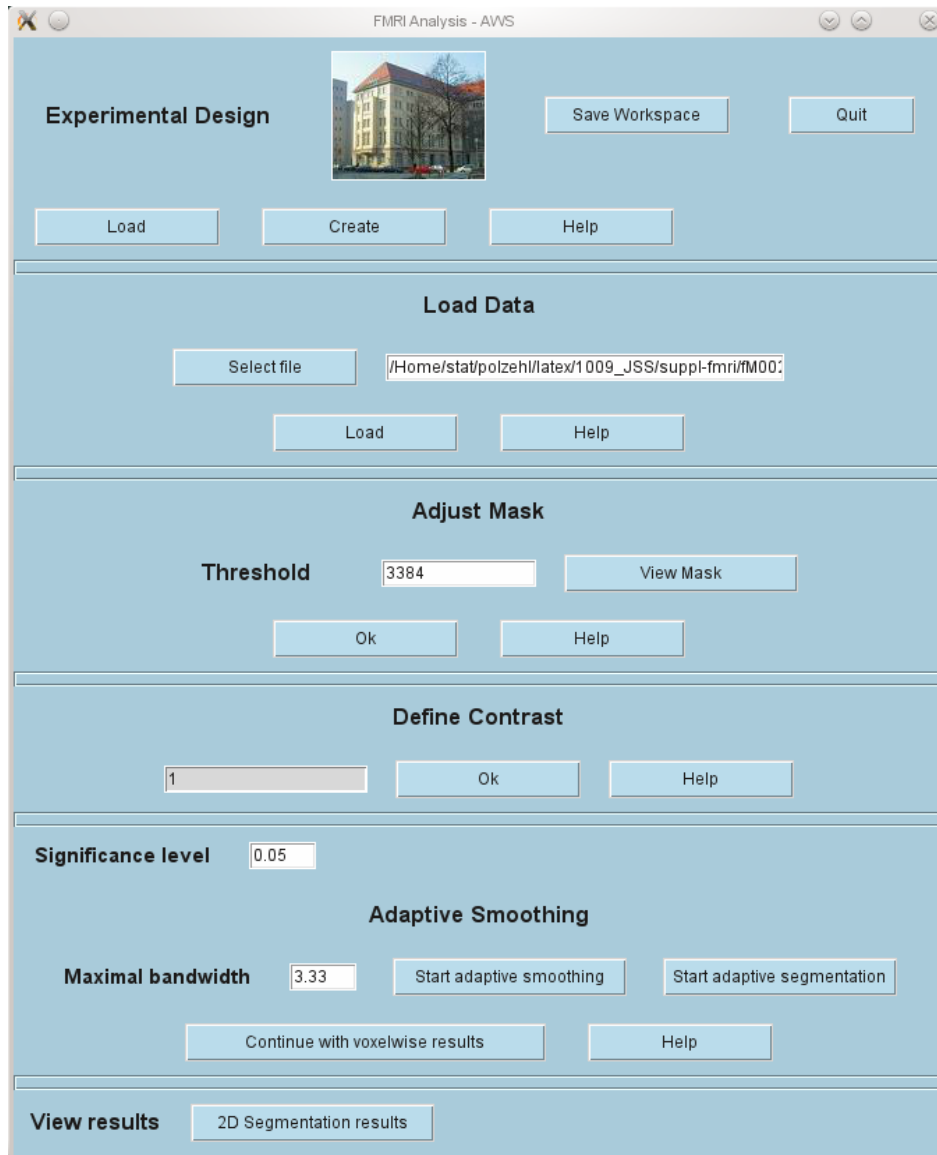
Figure 4: Snapshot of the **fmri** GUI after performing all steps.

Next a contrast must be specified as a vector separated by commas or blanks. Trailing zeros may be omitted. Given this information the SPM and the corresponding variance estimates are computed following the steps described in Section 2.3 using the function `fmri.lm`.

Finally a significance level (default: 0.05) needs to be specified. This is used within the adaptive segmentation algorithm (Section 3.2) or to specify the threshold for multiple test correction using RFT after structural adaptive smoothing. Finally adaptive smoothing (Section 3.1) or adaptive segmentation (Section 3.2) using a specified bandwidth may be performed. Figure 4 provides a snapshot of the GUI. Note that, if needed the GUI may be closed with or without saving the current results into the global environment of the R session. This enables one to continue the analysis using the full functionality of the package from the console.

In a last step results are visualized. The results view in 3D mode shows axial, sagittal
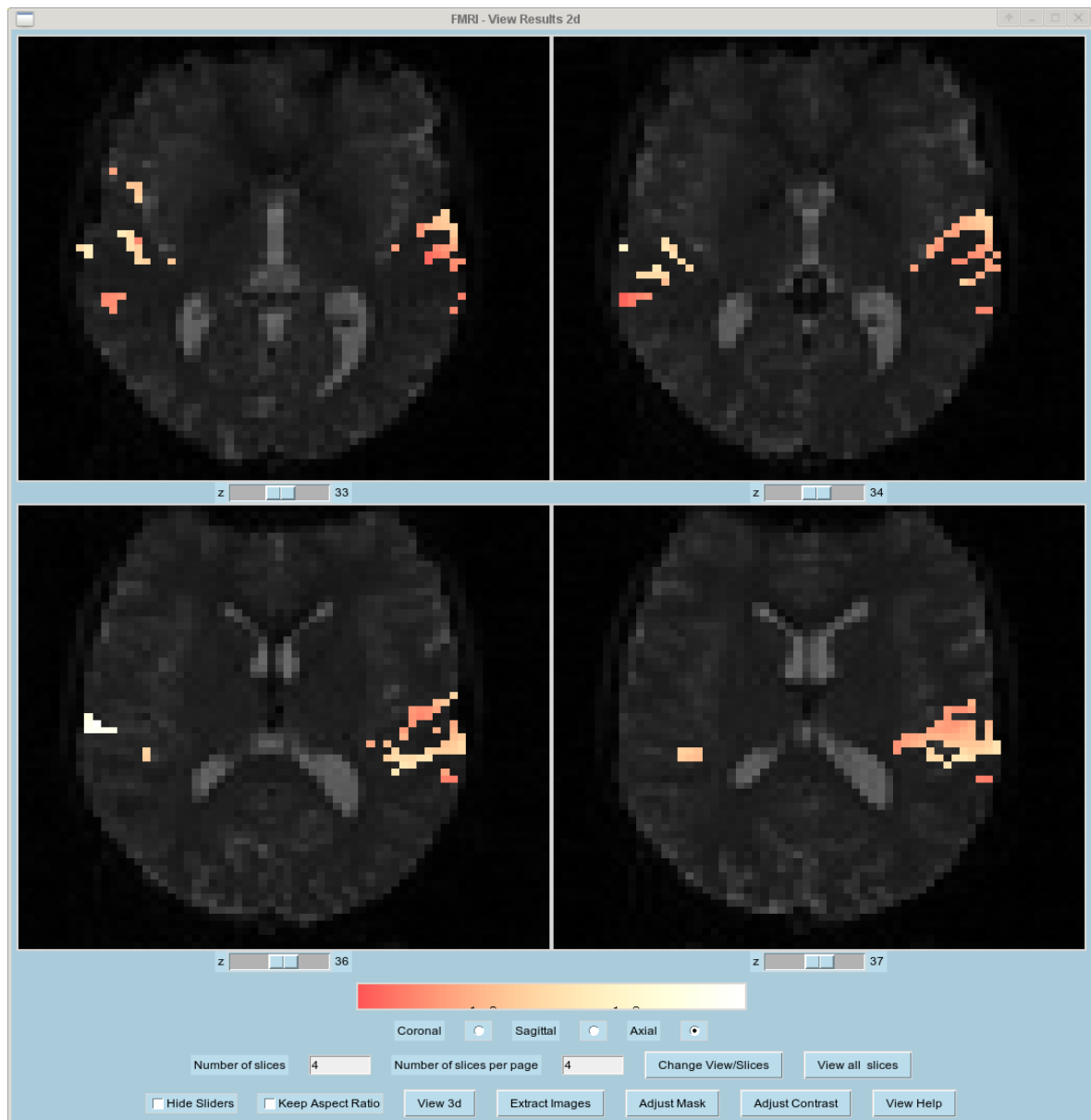
Figure 5: Snapshot of the results window of the GUI. This is the same as invoked by the `plot()` function.

and coronal slices with activations. The 2D view has more features (Figure 5). Sliders are available to navigate between the slices, while the view may be changed from axial to coronal, or sagittal. For convenience only four slices are shown by default. This may be changed using the number of slices and number of slices per page fields (`Change View/Slices` to apply the changes).

The `Extract images` button allows one to export images in JPEG or PNG format. Image processing is done using the package **adimpro** (Polzehl and Tabelow 2007a).

6. Conclusion

The R package **fmri** is a package for the analysis of single-subject fMRI data. Group analysis is currently not supported but will be included in a later version.

The package uses a linear model for the BOLD response. Based on the statistical parametric map noise reduction is performed by structural adaptive smoothing algorithms that reduce noise while not blurring the shape and extent of activated areas (Tabelow *et al.* 2006; Polzehl *et al.* 2010).

We demonstrated the usage of the main functions for fMRI analysis with two datasets which are publicly available. Finally we reported on the GUI which organizes the work-flow with slightly reduced functionality compared to the command-line.

Further development will include group analysis and a re-implementation using the advantages of S4 classes (Chambers 2008). One of the great strengths of the package is that it implements the complete fMRI analysis pipeline based on the general linear model (GLM) approach using new smoothing methods including signal detection. Other R packages for fMRI analysis are for example **AnalyzeFMRI** and **arf3DS4**.

Acknowledgments

References

Adler RJ (2000). "On Excursion Sets, Tube Formulae, and Maxima of Random Fields." *The Annals of Applied Probability*, **10**(1), 1–74.

Ashburner J, Chen CC, Flandin G, Henson R, Kiebel S, Kilner J, Litvak V, Moran R, Penny W, Stephan K, Hutton C, Glauche V, Mattout J, Phillips C (2008). *The **SPM8** Manual*. Functional Imaging Laboratory, Wellcome Trust Centre for Neuroimaging, Institute of Neurology, UCL, London. URL http://www.fil.ion.ucl.ac.uk/spm/.

Benjamini Y, Heller R (2007). "False Discovery Rates for Spatial Signals." *Journal of the American Statistical Association*, **102**(480), 1272–1281.

Benjamini Y, Hochberg Y (1995). "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing." *Journal of the Royal Statistical Society B*, **57**(1), 289–300.

Bordier C, Dojat M, Lafaye de Micheaux P (2011). "Temporal and Spatial Independent Component Analysis for fMRI Data Sets Embedded in the **AnalyzeFMRI** R Package." **44**(9), 1–24. URL http://www.jstatsoft.org/v44/i09/.

Chambers JM (2008). *Software for Data Analysis: Programming with R.* Springer-Verlag, New York.

Cheng K, Waggoner RA, Tanaka K (2001). "Human Ocular Dominance Columns as Revealed by High-Field Functional Magnetic Resonance Imaging." *Neuron*, **32**, 359–374.

Clayden J (2010). *RNiftyReg: Medical Image Registration Using the NiftyReg Library.* R package version 0.2.0, URL http://CRAN.R-project.org/package=RNiftyReg.

Cox RW (1996). "**AFNI**: Software for Analysis and Visualization of Functional Magnetic Resonance Neuroimages." *Computers and Biomedical Research*, **29**(3), 162–173.

Dalgaard P (2008). *Introductory Statistics with R.* 2nd edition. Springer-Verlag, New York.

Dümbgen L, Spokoiny V (2001). "Multiscale Testing of Qualitative Hypotheses." *The Annals of Statistics*, **29**(1), 124–152.

Ferreira da Silva A (2010). *cudaBayesreg: CUDA Parallel Implementation of a Bayesian Multilevel Model for fMRI Data Analysis.* R package version 0.3-9, URL http://CRAN.R-project.org/package=cudaBayesreg.

Ferreira da Silva AR (2011). "**cudaBayesreg**: Parallel Implementation of a Bayesian Multilevel Model for fMRI Data Analysis." *Journal of Statistical Software*, **44**(4), 1–24. URL http://www.jstatsoft.org/v44/i04/.

Friston K, Ashburner J, Kiebel S (eds.) (2007a). *Statistical Parametric Mapping: The Analysis of Functional Brain Images.* Academic Press.

Friston KJ, Ashburner J, Kiebel S, Nichols TE, Penny WD (eds.) (2007b). *Statistical Parametric Mapping: The Analysis of Functional Brain Images.* Elsevier.

Friston KJ, Holmes AP, Worsley KJ, Poline JB, Frith CD, Frackowiak RSJ (1995). "Statistical Parametric Maps in Functional Imaging: A General Linear Approach." *Human Brain Mapping*, **2**(4), 189–210.

Glover GH (1999). "Deconvolution of Impulse Response in Event-Related BOLD fMRI." *NeuroImage*, **9**(4), 416–429.

Haberg A, Kvistad KA, Unsgard G, Haraldseth O (2004). "Preoperative Blood Oxygen Level-Dependent Functional Magnetic Resonance Imaging in Patients with Primary Brain Tumors: Clinical Application and Outcome." *Neurosurgery*, **54**, 902–915.

Hayasaka S, Phan KL, Liberzon I, Worsley KJ, Nichols TE (2004). "Nonstationary Cluster-Size Inference with Random Field and Permutation Methods." *NeuroImage*, **22**(2), 676–687.

Henson JW, Gaviani P, Gonzalez RG (2005). "MRI in Treatment of Adult Gliomas." *Lancet Oncology*, **6**, 167–175.

Lazar NA (2008). *The Statistical Analysis of Functional MRI Data.* Statistics for Biology and Health. Springer-Verlag.

Marchini JL, Lafaye de Micheaux P (2010). ***AnalyzeFMRI***: *Functions for Analysis of fMRI Datasets Stored in the ANALYZE or NIFTI Format.* R package version 1.1-12, URL http://CRAN.R-project.org/package=AnalyzeFMRI.

Ogawa S, Lee TM, Kay AR, Tank DW (1990). "Brain Magentic Resonance Imaging with Contrast Dependent on Blood Oxygenation." *Proceedings of the National Academy of Sciences of the United States of America*, **87**(24), 9868–9872.

Ogawa S, Tank DW, Menon R, Ellermann JM, Kim S, Merkle H, Ugurbil K (1992). "Intrinsic Signal Changes Accompanying Sensory Stimulation: Functional Brain Mapping with Magnetic Resonance Imaging." *Proceedings of the National Academy of Sciences of the United States of America*, **89**(13), 5951–5955.

Polzehl J, Spokoiny V (2006). "Propagation-Separation Approach for Local Likelihood Estimation." *Probability Theory and Related Fields*, **135**(3), 335–362.

Polzehl J, Tabelow K (2007a). "Adaptive Smoothing of Digital Images: The R Package **adimpro**." *Journal of Statistical Software*, **19**(1), 1–17. URL http://www.jstatsoft.org/v19/i01/.

Polzehl J, Tabelow K (2007b). "**fmri**: A Package for Analyzing fMRI Data." R *News*, **7**(2), 13–17. URL http://CRAN.R-project.org/doc/Rnews/.

Polzehl J, Voss HU, Tabelow K (2010). "Structural Adaptive Segmentation for Statistical Parametric Mapping." *NeuroImage*, **52**(2), 515–523.

R Development Core Team (2010). R: *A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Resnick SI (1987). *Extreme Values, Regular Variation, and Point Processes.* Springer-Verlag.

Rorden C, Brett M (2000). "Stereotaxic Display of Brain Lesions." *Behavioural Neurology*, **12**(4), 191–200.

Smith SM, Jenkinson M, Woolrich MW, Beckmann CF, Behrens TEJ, Johansen-Berg H, Bannister PR, De Luca M, Drobnjak I, Flitney DE, Niazy RK, Saunders J, Vickers J, Zhang Y, De Stefano N, Brady JM, Matthews PM (2004). "Advances in Functional and Structural MR Image Analysis and Implementation as **FSL**." *NeuroImage*, **23**(Supplement 1), S208–S219.

Tabelow K, Piëch V, Polzehl J, Voss HU (2009). "High-Resolution fMRI: Overcoming the Signal-to-Noise Problem." *Journal of Neuroscience Methods*, **178**(2), 357–365.

Tabelow K, Polzehl J (2011). ***fmri***: *Analysis of fMRI Experiments.* R package version 1.4-6, URL http://CRAN.R-project.org/package=fmri.

Tabelow K, Polzehl J, Voss HU, Spokoiny V (2006). "Analyzing fMRI Experiments with Structural Adaptive Smoothing Procedures." *NeuroImage*, **33**(1), 55–62.

Vlieger EJ, Majoie CB, Leenstra S, den Heeten GJ (2004). "Functional Magnetic Resonance Imaging for Neurosurgical Planning in Neurooncology." *European Radiology*, **14**, 1143–1153.

Weeda WD (2010). *arf3DS4: Activated Region Fitting, fMRI Data Analysis (3D)*. R package version 2.4-7, URL http://CRAN.R-project.org/package=arf3DS4.

Weeda WD, de Vos F, Waldorp L, Grasman R, Huizenga H (2011). "**arf3DS4**: An Integrated Framework for Localization and Connectivity Analysis of fMRI Data." *Journal of Statistical Software*, **44**(14), 1–33. URL http://www.jstatsoft.org/v44/i14/.

Whitcher B (2010a). "CRAN Task View: Medical Image Analysis." Version 2010-11-30, URL http://CRAN.R-project.org/view=MedicalImaging.

Whitcher B (2010b). *oro.dicom: Rigorous – DICOM Input / Output*. R package version 0.2.7, URL http://CRAN.R-project.org/package=oro.dicom.

Whitcher B, Schmid V, Thornton A (2010). *oro.nifti: Rigorous – NIfTI Input / Output*. R package version 0.2.1, URL http://CRAN.R-project.org/package=oro.nifti.

Whitcher B, Schmid VJ, Thornton A (2011). "Working with the DICOM and NIfTI Data Standards in R." *Journal of Statistical Software*, **44**(6), 1–28. URL http://www.jstatsoft.org/v44/i06/.

Worsley KJ (1994). "Local Maxima and the Expected Euler Characteristic of Excursion Sets of χ^2, F and t Fields." *Advances in Applied Probability*, **26**(1), 13–42.

Worsley KJ (2005). "Spatial Smoothing of Autocorrelations to Control the Degrees of Freedom in fMRI Analysis." *NeuroImage*, **26**(2), 635–641.

Worsley KJ, Liao C, Aston JAD, Petre V, Duncan GH, Morales F, Evans AC (2002). "A General Statistical Analysis for fMRI Data." *NeuroImage*, **15**(1), 1–15.

Affiliation:

Karsten Tabelow, Jörg Polzehl
Weierstrass Institute for Applied Analysis and Stochastics
Mohrenstr. 39
10117 Berlin, Germany
E-mail: karsten.tabelow@wias-berlin.de,
        joerg.polzehl@wias-berlin.de
URL: http://www.wias-berlin.de/projects/matheon_a3/