

MULTI-SCALE MINING OF FMRI DATA WITH HIERARCHICAL STRUCTURED SPARSITY

RODOLPHE JENATTON ^{*}, ALEXANDRE GRAMFORT [†], VINCENT MICHEL [†], GUILLAUME OBOZINSKI ^{*}, EVELYN EGER [‡], FRANCIS BACH ^{*}, AND BERTRAND THIRION [†]

Abstract. Inverse inference, or “*brain reading*”, is a recent paradigm for analyzing functional magnetic resonance imaging (fMRI) data, based on pattern recognition and statistical learning. By predicting some cognitive variables related to brain activation maps, this approach aims at decoding brain activity. Inverse inference takes into account the multivariate information between voxels and is currently the only way to assess how precisely some cognitive information is encoded by the activity of neural populations within the whole brain. However, it relies on a prediction function that is plagued by the curse of dimensionality, since there are far more features than samples, *i.e.*, more voxels than fMRI volumes. To address this problem, different methods have been proposed, such as, among others, univariate feature selection, feature agglomeration and regularization techniques. In this paper, we consider a sparse hierarchical structured regularization. Specifically, the penalization we use is constructed from a tree that is obtained by spatially-constrained agglomerative clustering. This approach encodes the spatial structure of the data at different scales into the regularization, which makes the overall prediction procedure more robust to inter-subject variability. The regularization used induces the selection of spatially coherent predictive brain regions simultaneously at different scales. We test our algorithm on real data acquired to study the mental representation of objects, and we show that the proposed algorithm not only delineates meaningful brain regions but yields as well better prediction accuracy than reference methods.

Key words. brain reading, structured sparsity, convex optimization, sparse hierarchical models, inter-subject validation, proximal methods.

AMS subject classifications. -

1. Introduction. Functional magnetic resonance imaging (or fMRI) is a widely used functional neuroimaging modality. Modeling and statistical analysis of fMRI data are commonly done through a linear model, called general linear model (GLM) in the community, that incorporates information about the different experimental conditions and the dynamics of the hemodynamic response in the design matrix. The experimental conditions are typically modelled by the type of stimulus presented, *e.g.*, visual and auditory stimulation, which are included as regressors in the design matrix. The resulting model parameters—one coefficient per voxel and regressor—are known as *activation maps*. They represent the local influence of the different experimental conditions on fMRI signals at the level of individual voxels. The most commonly used approach to analyze these activation maps is called classical inference. It relies on mass-univariate statistical tests (one for each voxel), and yields so-called statistical parametric maps (SPMs) [13]. Such maps are useful for functional brain mapping, but classical inference has some limitations: it suffers from multiple comparisons issues and it is oblivious of the multivariate structure of fMRI data. Such data exhibit natural correlations between neighboring voxels forming clusters with different sizes and shapes, and also between distant but functionally connected brain regions.

To address these limitations, an approach called inverse inference (or “brain-reading”) [9, 8] was recently proposed. Inverse inference relies on pattern recognition tools and statistical learning methods to explore fMRI data. Based on a set of activation maps, inverse inference estimates a function that can then be used to predict a target (typically, a variable representing a perceptual, cognitive or behavioral parameter) for a new set of images. The challenge is to

^{*}INRIA Rocquencourt - Sierra Project-Team, Laboratoire d’Informatique de l’Ecole Normale Supérieure, INRIA/ENS/CNRS UMR 8548 (firstname.lastname@inria.fr).

[†]INRIA Saclay - Parietal Project-Team, CEA Neurospin. (firstname.lastname@inria.fr).

[‡]INSERM U562, France - CEA/DSV/I2BM/Neurospin/Unicog (evelyn.eger@cea.fr).

⁰A preliminary version of this work appeared in [20].

capture the correlation structure present in the data in order to improve the performance of the mapping learnt, which is measured through the resulting prediction accuracy. Many standard statistical learning approaches have been used to construct prediction functions, among them kernel machines (SVM, RVM) [37] or discriminant analysis (LDA, QDA) [16]. For the application considered in this paper, earlier performance results [8, 25] indicate that we can restrict ourselves to mappings that are linear functions of the data.

Throughout the paper, we shall consider a training set composed of n pairs $(\mathbf{x}, y) \in \mathbb{R}^p \times \mathcal{Y}$, where \mathbf{x} denotes a p -dimensional fMRI signal (p voxels) and y stands for the target we try to predict. In the experiments we carry out in Section 5, we will encounter both the regression and the multi-class classification settings, where \mathcal{Y} denotes respectively the set of real numbers and a finite set of integers. In this paper, we aim at learning a weight vector $\mathbf{w} \in \mathbb{R}^p$ and an intercept $b \in \mathbb{R}$ such that the prediction of y can be based on the value of $\mathbf{w}^\top \mathbf{x} + b$. This is the case for the linear regression and logistic regression models that we use in Section 5. It is useful to rewrite these quantities in matrix form; more precisely, we denote by $\mathbf{X} \in \mathbb{R}^{n \times p}$ the design matrix assembled from n fMRI data points and by $\mathbf{y} \in \mathbb{R}^n$ the corresponding n targets. In other words, each row of \mathbf{X} is a p -dimensional sample, *i.e.*, an activation map of p voxels related to one stimulus presentation.

Learning the parameters (\mathbf{w}, b) remains challenging since the number of features (10^4 to 10^5 voxels) exceeds by far the number of samples (a few hundreds of images). The prediction function is therefore prone to the phenomenon of overfitting in which the learning set is predicted precisely whereas the algorithm provides very inaccurate predictions on new samples (the test set). To address this issue, *dimensionality reduction* attempts to find a low dimensional subspace that concentrates as much of the predictive power of the original set as possible for the problem at hand.

Feature selection is a natural approach to perform dimensionality reduction in fMRI, since reducing the number of voxels potentially allows to identify a predictive region of the brain. This corresponds to discarding some columns of \mathbf{X} . This feature selection can be univariate, *e.g.*, analysis of variance (ANOVA) [26], or multivariate. While univariate methods ignore joint information between features, multivariate approaches are more adapted to inverse inference since they extract predictive patterns from the data as a whole. However, due to the huge number of possible patterns, these approaches suffer from combinatorial explosion, and some costly suboptimal heuristics (*e.g.*, recursive feature elimination [15, 28]) can be used. That is why ANOVA is usually preferred in fMRI. Alternatively, two more adapted solutions have been proposed: *regularization* and *feature agglomeration*.

Regularization is a way to encode a priori knowledge about the weight vector \mathbf{w} . Possible regularizers can promote for example spatial smoothness or sparsity which is a natural assumption for fMRI data. Indeed, only a few brain regions are assumed to be significantly activated during a cognitive task. Previous contributions on fMRI-based inverse inference include [4, 35, 36, 43]. They can be presented through the following minimization problem:

$$\min_{(\mathbf{w}, b) \in \mathbb{R}^{p+1}} \mathcal{L}(\mathbf{y}, \mathbf{X}, \mathbf{w}, b) + \lambda \Omega(\mathbf{w}) \quad \text{with } \lambda \geq 0, \quad (1.1)$$

where $\lambda \Omega(\mathbf{w})$ is the regularization term, typically a non-Euclidean norm, and the fit to the data is measured through a convex loss function $(\mathbf{w}, b) \mapsto \mathcal{L}(\mathbf{y}, \mathbf{X}, \mathbf{w}, b) \in \mathbb{R}_+$. The choice of the loss function will be made more specific and formal in the next sections. The coefficient of regularization λ balances the loss and the penalization term. In this notation, a common regularization term in inverse inference is the so-called *Elastic net* [45, 14], which

is a combined ℓ_1 and ℓ_2 penalization:

$$\lambda\Omega(\mathbf{w}) = \lambda_1\|\mathbf{w}\|_1 + \lambda_2\|\mathbf{w}\|_2^2 = \sum_{j=1}^p \{\lambda_1|\mathbf{w}_j| + \lambda_2\mathbf{w}_j^2\}. \quad (1.2)$$

For the square loss, when setting λ_1 to 0, the model is called ridge regression, while when $\lambda_2 = 0$ it is known as Lasso [39] or basis pursuit [5]. The essential shortcoming of the Elastic net is that it does not take into account the spatial structure of the data, which is crucial in this context [30]. Indeed, due to the intrinsic smoothing of the complex metabolic pathway underlying the difference of blood oxygenation measured with fMRI [40], statistical learning approaches should be informed by the 3D grid structure of the data.

In order to achieve dimensionality reduction, while taking into account the spatial structure of the data, one can resort to *feature agglomeration*. It constructs new features, called *parcels*, by averaging neighboring voxels exhibiting similar activations. The advantage of agglomeration is that no information is discarded a priori and that it is reasonable to hope that averaging might reduce noise. Although, this approach has been successfully used in previous work for brain mapping [12, 38], it often does not consider the supervised information (*i.e.*, the target \mathbf{y}) while constructing the parcels. A recent approach has been proposed to address this issue, using a supervised greedy top-down exploration of a tree obtained by hierarchical clustering [29]. This greedy approach has proven to be effective especially for inter-subject analyses, *i.e.*, when the training and the evaluation sets are related to different subjects. In this context, methods need to be robust to intrinsic spatial variations that exist across subjects: despite being co-registered into a common space, some variability remains between subjects, which implies that there is no perfect voxel-to-voxel correspondence between volumes. As a result, the performances of traditional voxel-based methods are strongly affected. Therefore, averaging in the form of parcels is a good way to cope with inter-subject variability. This greedy approach is nonetheless suboptimal, as it explores only a subpart of the whole tree.

Based on these considerations, we propose to integrate the multi-scale spatial structure of the data *within* the regularization term Ω , while preserving convexity in the optimization. This notably guarantees global optimality and stability of the obtained solutions. To this end, we design a sparsity-inducing penalty that is directly built from the hierarchical structure of the spatial model obtained by Ward’s algorithm [41]. Such a penalty has already been successfully applied in several contexts, *e.g.*, in bioinformatics, to exploit the tree structure of gene networks for multi-task regression [24], and also for topic models and image inpainting [22].

We summarize here the contributions of our paper:

- We explain how the multi-scale spatial structure of fMRI data can be taken into account in the context of inverse inference through the combination of a spatially constrained hierarchical clustering procedure and a sparse hierarchical regularization.
- We provide a convex formulation of the problem and propose an efficient optimization procedure.
- We conduct an experimental comparison of several algorithms and formulations on fMRI data and illustrate the ability of the proposed method to localize in space and in scale some brain regions involved in the processing of visual stimuli.

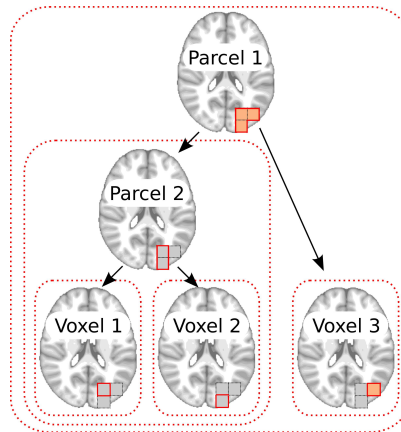
The rest of the paper is organized as follows: we first present the concept of structured sparsity-inducing regularization and then describe the different regression/classification formulations we are interested in. After exposing how we handle the resulting large-scale convex optimization problems thanks to proximal methods, we validate our approach on both a synthetic setting and a real dataset.

2. Combining agglomerative clustering with sparsity inducing regularizers. Hierarchical clustering allows to construct a tree-structured hierarchy of features on top of the original voxels features. Moreover, the underlying voxels corresponding to each of these features correspond to localized spatial patterns on the brain of the form we hope to retrieve [6]. Instead of selecting features in the tree greedily, we propose to cast the feature selection problem as supervised learning problem of the form (1.1). It is natural to require of the regularizer Ω that it should respect the tree structure of the hierarchy so as to induce the selection of localized patterns.

2.1. Constructing the sparsity-inducing norm. The structured sparsity-inducing term Ω is built from the result of the hierarchical clustering of the voxels. The latter yields a hierarchy of *clusters* represented as a tree \mathcal{T} (or dendrogram) [23]. The root of the tree is the unique cluster that gathers all the voxels, while the leaves are the clusters with a single voxel. Among different *hierarchical agglomerative clustering* procedures, we use the variance-minimizing approach of Ward’s algorithm [41], since it minimizes the loss of information at each step of clustering. In short, two *clusters* are merged if the resulting parcellation minimizes the sum of squared differences within all *clusters* (also known as *inertia criterion*).

In order to take into account the spatial information, we also add connectivity constraints in the hierarchical clustering algorithm, so that only neighboring clusters can be merged together. The resulting clusters are thus called *parcels*. Each node of the tree \mathcal{T} either corresponds to a voxel if it is a leaf, or defines a *parcel*, as the union of its children’s clusters of voxels (see Figure 2.1).

FIG. 2.1. Example of a tree \mathcal{T} when $p = 5$, with three voxels and two parcels. The parcel 2 is defined as the averaged intensity of the voxels $\{1, 2\}$, while the parcel 1 is obtained by averaging the parcel 2 and voxel 3. In red dashed lines are represented the five groups of variables that compose \mathcal{G} . For instance, if the group containing the parcel 2 is set to zero, the voxels $\{1, 2\}$ are also (and necessarily) zeroed out. Best seen in color.



We now consider the augmented space of variables (also known as features), formed by not only the voxels, but also by the parcels. This approximately doubles the number of features of the fMRI signals. In other words, p does not denote the number of voxels anymore, but instead, the total number of nodes of \mathcal{T} .¹ In the following, the level of activation of each parcel is (recursively) defined by the averaged intensity of the voxels it is composed of (*i.e.*, local averages) [12, 38]. This produces a multi-scale representation of the fMRI data that becomes increasingly invariant to spatial shifts of the encoding regions within the brain volume.

More formally, if j is a node of \mathcal{T} and P_j stands for the set of voxels of the corresponding parcel (*i.e.*, the set of leaves of the subtree rooted at node j), we consider the mean of the

¹We can then identify nodes (and parcels) of \mathcal{T} with indices in $\{1, \dots, p\}$.

parcel that we denote by $\langle \mathbf{x}_{P_j} \rangle$. In this notation, the linear model we use is of the form

$$f_{\mathbf{w}}(\mathbf{x}) = \mathbf{w}^\top \tilde{\mathbf{X}} = \sum_{j \in \mathcal{T}} \mathbf{w}_j \langle \mathbf{x}_{P_j} \rangle = \sum_{i \in V} \left[\sum_{j \in A(i)} \frac{\mathbf{w}_j}{|P_j|} \right] \mathbf{x}_i,$$

where $A(i)$ is the set of ancestors of a node i in \mathcal{T} (including itself), and V corresponds to the leaves of the tree. To lighten notations, in the remainder of the paper, we will denote by \mathbf{X} instead of $\tilde{\mathbf{X}}$ the matrix of features from the augmented space.

In the perspective of inter-subject validation, the augmented space of variables can be exploited in the following way: since the information of single voxels may be unreliable, *the deeper the node in \mathcal{T} , the more variable the corresponding parcel's intensity is likely to be across subjects*. This property suggests that, while looking for sparse solutions of (1.1), we should preferentially select the variables near the root of \mathcal{T} , before trying to access smaller parcels located further down in \mathcal{T} .

Traditional sparsity-inducing penalties, *e.g.*, the ℓ_1 -norm $\Omega(\mathbf{w}) = \sum_{j=1}^p |\mathbf{w}_j|$, yield sparsity at the level of single variables \mathbf{w}_j , disregarding potential structures—for instance, spatial—existing between larger subsets of variables. We leverage here the concept of *structured sparsity* where Ω penalizes some predefined subsets, or *groups*, of variables that reflect prior information about the problem at hand [1, 17, 19, 18]. In particular, we follow [44] that first introduced hierarchical sparsity-inducing penalties. Given a node j of \mathcal{T} , we denote by $g_j \subseteq \{1, \dots, p\}$ the set of indices that record all the descendants of j in \mathcal{T} , including itself. In other words, g_j contains the indices of the subtree rooted at j ; see Figure 2.1. If we now denote by \mathcal{G} the set of all g_j , $j \in \{1, \dots, p\}$, that is, $\mathcal{G} \triangleq \{g_1, \dots, g_p\}$, we can define our hierarchical penalty as

$$\Omega(\mathbf{w}) \triangleq \sum_{g \in \mathcal{G}} \|\mathbf{w}_g\|_2 \triangleq \sum_{g \in \mathcal{G}} \left[\sum_{j \in g} \mathbf{w}_j^2 \right]^{1/2}. \quad (2.1)$$

As shown in [19], Ω is a norm, and it promotes sparsity at the level of groups $g \in \mathcal{G}$, in the sense that it acts as a ℓ_1 -norm on the vector $(\|\mathbf{w}_g\|_2)_{g \in \mathcal{G}}$. Regularizing by Ω therefore causes some $\|\mathbf{w}_g\|_2$ (and equivalently \mathbf{w}_g) to be zeroed out for some $g \in \mathcal{G}$. Moreover, since the groups $g \in \mathcal{G}$ represent rooted subtrees of \mathcal{T} , this implies that if one node/parcel $j \in g$ is set to zero by Ω , the same occurs for all its descendants [44]. To put it differently, *if one parcel is selected, then all the ancestral parcels in \mathcal{T} will also be selected*. This property is in accordance with our concern of robustness with respect to voxel misalignments between subjects, since large parcels are considered before smaller ones.

The family of norms with the previous property is actually slightly larger and we consider throughout the paper norms Ω of the form [44]:

$$\Omega(\mathbf{w}) \triangleq \sum_{g \in \mathcal{G}} \eta_g \|\mathbf{w}_g\|, \quad (2.2)$$

where $\|\mathbf{w}_g\|$ denotes either the ℓ_2 -norm $\|\mathbf{w}_g\|_2$ or the ℓ_∞ -norm $\|\mathbf{w}_g\|_\infty \triangleq \max_{j \in g} |\mathbf{w}_j|$ and $(\eta_g)_{g \in \mathcal{G}}$ are (strictly) positive weights that can compensate for the fact that some features are overpenalized as a result of being included in a larger number of groups than others. In light of the results from [22], we will see in Section 4 that a large class of optimization problems regularized by Ω —as defined in (2.2)—can be solved efficiently.

3. Supervised learning framework. In this section, we introduce the formulations we consider in our experiments. As further discussed in Section 5, the target y we try to predict corresponds to (discrete) sizes of objects, *i.e.*, a one-dimensional *ordered* variable. It is therefore sensible to address this prediction task from both a regression and a classification viewpoint.

3.1. Regression. In this first setting, we naturally consider the square loss function, so that problem (1.1) can be reduced to

$$\min_{\mathbf{w} \in \mathbb{R}^p} \frac{1}{2n} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2 + \lambda \Omega(\mathbf{w}) \quad \text{with } \lambda \geq 0.$$

Note that in this case, we have omitted the intercept b since we can center the vector \mathbf{y} and the columns of \mathbf{X} instead.

3.2. Classification. We now look at our prediction task from a multi-class classification viewpoint. Specifically, we assume that \mathcal{Y} is a finite set of integers $\{1, \dots, c\}$, $c > 2$, and consider both multi-class and “one-versus-all” strategies [34]. We need to slightly extend the formulation (1.1): To this end, we introduce the weight matrix $\mathbf{W} \triangleq [\mathbf{w}^1, \dots, \mathbf{w}^c] \in \mathbb{R}^{p \times c}$, composed of c weight vectors, along with a vector of intercepts $\mathbf{b} \in \mathbb{R}^c$.

A standard way of addressing multi-class classification problems consists in using a multi-logit model, also known as multinomial logistic regression (see, e.g., [16] and references therein). In this case, class-conditional probabilities are modeled for each class by a softmax function and leads to

$$\min_{\substack{\mathbf{W} \in \mathbb{R}^{p \times c} \\ \mathbf{b} \in \mathbb{R}^c}} \frac{1}{n} \sum_{i=1}^n \log \left[\sum_{k=1}^c e^{\mathbf{x}_i^\top (\mathbf{w}^k - \mathbf{w}^{y_i}) + \mathbf{b}_k - \mathbf{b}_{y_i}} \right] + \lambda \sum_{k=1}^c \Omega(\mathbf{w}^k).$$

Whereas the regularization term is separable with respect to the different weight vectors \mathbf{w}^k , the loss function induces a coupling in the columns of \mathbf{W} . As a result, the optimization has to be carried out over the entire matrix \mathbf{W} .

In Section 5, we consider another multi-class classification scheme. The “one-versus-all” strategy (OVA) consists in training c different (real-valued) binary classifiers, each one being trained to distinguish the examples in a single class from the observations in all remaining classes. In order to classify a new example, among the c classifiers, the one which outputs the largest (most positive) value is chosen. In this framework, we consider binary classifiers built from both the square and the logistic loss functions. If we denote by $\bar{\mathbf{Y}} \in \{-1, 1\}^{n \times c}$ the indicator response matrix defined as $\bar{\mathbf{Y}}_i^k \triangleq 1$ if $y_i = k$ and -1 otherwise, we obtain

$$\min_{\mathbf{W} \in \mathbb{R}^{p \times c}} \frac{1}{2n} \sum_{k=1}^c \|\bar{\mathbf{Y}}^k - \mathbf{X}\mathbf{w}^k\|_2^2 + \lambda \sum_{k=1}^c \Omega(\mathbf{w}^k),$$

and

$$\min_{\substack{\mathbf{W} \in \mathbb{R}^{p \times c} \\ \mathbf{b} \in \mathbb{R}^c}} \frac{1}{n} \sum_{i=1}^n \sum_{k=1}^c \log \left[1 + e^{-\bar{\mathbf{Y}}_i^k (\mathbf{x}_i^\top \mathbf{w}^k + \mathbf{b}_k)} \right] + \lambda \sum_{k=1}^c \Omega(\mathbf{w}^k).$$

By invoking the same arguments as in Section 3.1, the vector of intercepts \mathbf{b} is again omitted in the above problem with the square loss. The formulations we reviewed in this section can be solved efficiently within the same optimization framework we now introduce.

4. Optimization. The convex minimization problem (1.1) is challenging, since the penalty Ω as defined in (2.2) is non-smooth and the number of variables to deal with is large (about $p \approx 10^5$ voxels in the following experiments). To this end, we resort to *proximal methods* (see, e.g., [2, 7, 31, 42]). In a nutshell, these methods can be seen as a natural extension of gradient-based techniques when the objective function to minimize has an amenable non-smooth part. They have increasingly drawn the attention of a broad research community because of their convergence rates (optimal within the class of first-order techniques) and their

ability to deal with large non-smooth convex problems. We assume from now on that the convex loss function $\mathcal{L}(\mathbf{y}, \mathbf{X}, \cdot)$ is differentiable with Lipschitz-continuous gradient, which notably covers the cases of the square and simple/multinomial logistic functions (introduced in Section 3).

The simplest version of this class of methods linearizes at each iteration the function $\mathcal{L}(\mathbf{y}, \mathbf{X}, \cdot)$ around the current estimate \mathbf{w}_0 ,² and this estimate is then updated as the (unique by strong convexity) solution of the *proximal problem*:

$$\min_{\mathbf{w} \in \mathbb{R}^p} \mathcal{L}(\mathbf{y}, \mathbf{X}, \mathbf{w}_0) + (\mathbf{w} - \mathbf{w}_0)^\top \nabla \mathcal{L}_{\mathbf{w}}(\mathbf{y}, \mathbf{X}, \mathbf{w}_0) + \lambda \Omega(\mathbf{w}) + \frac{L}{2} \|\mathbf{w} - \mathbf{w}_0\|_2^2.$$

The quadratic term keeps the update in a neighborhood where $\mathcal{L}(\mathbf{y}, \mathbf{X}, \mathbf{w}_0)$ is close to its linear approximation, and $L > 0$ is a parameter which is an upper bound on the Lipschitz constant of the gradient of \mathcal{L} . This problem can be equivalently rewritten as:

$$\min_{\mathbf{w} \in \mathbb{R}^p} \frac{1}{2} \left\| \mathbf{w}_0 - \frac{1}{L} \nabla \mathcal{L}_{\mathbf{w}}(\mathbf{y}, \mathbf{X}, \mathbf{w}_0) - \mathbf{w} \right\|_2^2 + \frac{\lambda}{L} \Omega(\mathbf{w}). \quad (4.1)$$

Solving efficiently and exactly this problem is crucial to enjoy the fastest convergence rates of proximal methods. In addition, when the non-smooth term Ω is not present, the previous proximal problem exactly leads to the standard gradient update rule. In simple settings, the solution of problem (4.1) is given in closed form: For instance, when the regularization Ω is chosen to be the ℓ_1 -norm, we get back the well-known soft-thresholding operator [10].

The work of [22] recently showed that the proximal problem (4.1) could be solved efficiently and exactly with Ω as defined in (2.2). The underlying idea of this computation is to solve a *well-ordered* sequence of simple proximal problems associated with each of the terms $\|\mathbf{w}_g\|$ for $g \in \mathcal{G}$. We refer the interested readers to [21] for further details.

In our experiments, we will use the accelerated proximal gradient scheme (FISTA) taken from [2], which is a similar procedure as the one described above, except that the proximal problem (4.1) is not solved for the current estimate, but for an auxiliary sequence of points that are linear combinations of past estimates.³ In terms of computational complexity, such proximal schemes are guaranteed to be ε close to the optimal objective function in $O(\sqrt{L/\varepsilon})$ iterations [2, 31]. The cost of each iteration is dominated by the computation of the gradient (e.g., $O(np)$ for the square loss) and the proximal operator, whose time complexity is linear, or close to linear, in p for the tree-structured regularization [21].

5. Experiments and results. We now present experimental results on simulated data and real fMRI data.

5.1. Simulations. In order to illustrate the proposed method, the hierarchical regularization with the ℓ_2 -norm and $\eta_g = 1$ for all g was applied in a regression setting on a small two-dimensional simulated dataset consisting of 300 square images (40×40 pixels i.e. $\mathbf{X} \in \mathbb{R}^{300 \times 1600}$). The weight vector \mathbf{w} used in the simulation— itself an image of the same dimension— is presented in Fig. 5.1-a. It consists of three localized regions of two different sizes that are predictive of the output. The images $\mathbf{x}^{(i)}$ are sampled so as to obtain a correlation structure which mimics fMRI data. Precisely, each image $\mathbf{x}^{(i)}$ was obtained by smoothing a completely random image — where each pixel was drawn i.i.d from a normal distribution — with a Gaussian kernel, which introduces spatial correlations between neighboring pixels. Subsequently, correlations between the regions corresponding to the three

²For simplicity and clarity of the presentation, we do not consider the optimization of the intercept that we let unregularized in all our experiments.

³The Matlab/C++ implementation we use is available at <http://www.di.ens.fr/willow/SPAMS/>.

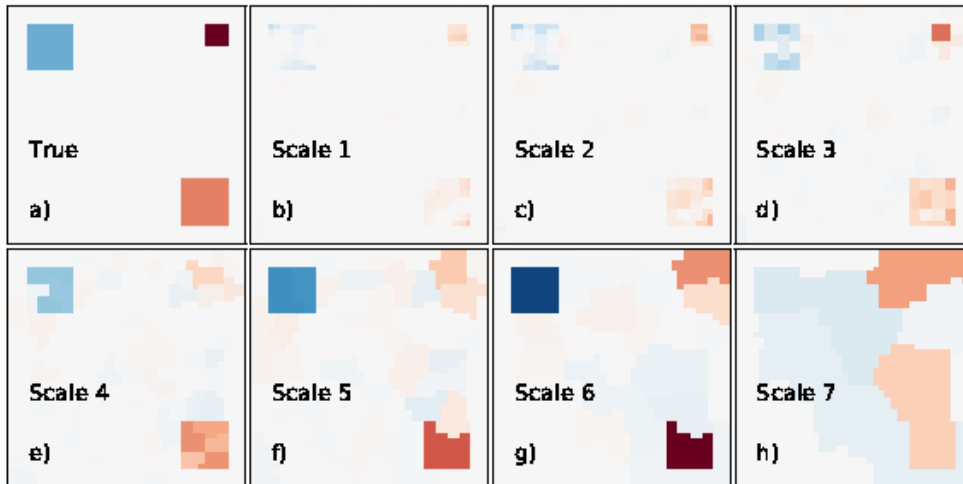


FIG. 5.1. *Weights estimated in the simulation study. The true coefficients are presented in a) and the estimated weights at different scales, i.e., different depths in the tree, are presented in b)-h). The results are best seen in color.*

patterns were introduced in order to simulate co-activations between different brain regions (0.3 correlation between the two bigger patterns, and -0.2 correlation between the smallest and lower-corner patterns).

The choice of the weights and of the correlation introduced in images aim at illustrating how the hierarchical regularization estimates weights at different resolutions in the image. The targets were simulated by forming $\mathbf{w}^\top x^{(i)}$ corrupted with an additive white noise (SNR=10dB). The loss used was the square loss as detailed in Section 3.1. The regularization parameter was estimated with two-fold cross-validation (150 images per fold) on a logarithmic grid of 30 values between 10^3 and 10^{-3} .

The weights estimated are presented in Fig. 5.1 at different scales, *i.e.*, different depths in the tree. It can be observed that all three patterns are present in the weight vector but at different depth in the tree. The small activation in the top-right hand corner shows up mainly in scale 3 while the bigger patterns appear higher in the tree in scales 5 and 6. This simulation clearly illustrates the ability of the method to capture informative spatial patterns at different scales. We now present results on real data.

5.2. Description on the data. We apply the different methods to analyze the data of ten subjects from an fMRI study originally designed to investigate object coding in high-level visual cortex (see [11] for details). During the experiment, twelve healthy volunteers viewed objects of two categories (each one of the two categories is used in half of the subjects) with four different exemplars in each category. Each exemplar was presented at three different sizes (yielding 12 different experimental conditions per subject). Each stimulus was presented four times in each of the six sessions. We averaged data from the four repetitions, resulting in a total of $n = 72$ images by subject (one image of each stimulus by session). Functional images were acquired on a 3-T MR system with eight-channel head coil (Siemens Trio, Erlangen, Germany) as T2*-weighted echo-planar image (EPI) volumes. Twenty transverse slices were obtained with a repetition time of 2s (echo time, 30ms; flip angle, 70° ; $2 \times 2 \times 2$ -mm voxels; 0.5-mm gap). Realignment, normalization to MNI space, and GLM fit were performed with the SPM5 software⁴. In the GLM, the time course of each

⁴<http://www.fil.ion.ucl.ac.uk/spm/software/spm5>.

of the 12 stimuli convolved with a standard hemodynamic response function was modeled separately, while accounting for serial auto-correlation with an AR(1) model and removing low-frequency drift terms with a high-pass filter with a cut-off of 128s. In the present work we used the resulting session-wise parameter estimate images. All the analysis are performed on the whole brain volume.

The four different exemplars in each of the two categories were pooled, leading to images labeled according to the three possible sizes of the object. By doing so, we are interested in finding discriminative information to predict the size of the presented object.

This can be reduced to either a regression problem in which our goal is to predict a simple scalar factor (size or scale of the presented object), or a three-category classification problem, each size corresponding to a category. We perform an inter-subject analysis on the sizes both in regression and classification settings. This analysis relies on subject-specific fixed-effects activations, *i.e.*, for each condition, the six activation maps corresponding to the six sessions are averaged together. This yields a total of 12 images per subject, one for each experimental condition. The dimensions of the real data set are $p \approx 7 \times 10^4$ and $n = 120$ (divided into three different sizes). We evaluate the performance of the method by cross-validation with a natural data splitting, *leave-one-subject-out*. Each fold consists of 12 volumes. The parameter λ of all methods is optimized over a grid of 30 values of the form 2^k , with a nested leave-one-subject-out cross-validation on the training set. The exact scaling of the grid varies for each model to account for different Ω .

5.3. Methods involved in the comparisons. In addition to considering standard ℓ_1 - and squared ℓ_2 -regularizations in both our regression and multi-class classification tasks, we compare various methods that we now review.

First of all, when the regularization Ω as defined in (2.2) is employed, we consider three settings of values for $(\eta_g)_{g \in \mathcal{G}}$ which leverage the tree structure \mathcal{T} . More precisely, we set $\eta_g = \rho^{\text{depth}(g)}$ for g in \mathcal{G} , with $\rho \in \{0.5, 1, 1.5\}$ and where $\text{depth}(g)$ denotes the depth of the root of the group g in \mathcal{T} . In other words, the larger ρ , the more averse we are to selecting small (and variable) parcels located near the leaves of \mathcal{T} .

The greedy approach from [29] is included in the comparisons, for both the regression and classification tasks. It relies on a top-down exploration of the tree \mathcal{T} . In short, starting from the root parcel that contains all the voxels, we choose at each step the split of the parcel that yields the highest prediction score. The exploration step is performed until a given number of parcels is reached, and yields a set of nested parcellations with increasing complexity. Similarly to a model selection step, we chose the best parcellation among those found in the exploration step. The selected parcellation is thus used on the test set. In the regression setting, this approach is combined with Bayesian ridge regression, while it is associated with a linear support vector machine for the classification task (whose value of C is found by nested cross-validation in $\{0.01, 0.1, 1\}$).

5.3.1. Regression setting. In order to evaluate whether the level of sparsity is critical in our analysis, we implemented a reweighted ℓ_1 -scheme [3]. In this case, sparsity is encouraged more aggressively as a multi-stage convex relaxation of a concave penalty. Specifically, it consists in using iteratively a weighted ℓ_1 -norm, whose weights are determined by the solution of previous iteration.

To better understand the added value of the hierarchical norm (2.2) over unstructured penalties, we consider another variant of weighted ℓ_1 -norm, this time defined in the augmented space of features. The weights are manually set and reflect the underlying tree structure \mathcal{T} . By analogy with the choice of $(\eta_g)_{g \in \mathcal{G}}$ made for the tree-structured regularization,

we take exponential weights depending on the depth of the variable j , with $\rho = 1.5$.⁵ We also tried weights $(\eta_g)_{g \in \mathcal{G}}$ that are linear with respect to the depths, but those led to worse results. We now turn to the models taking part in the classification task.

5.3.2. Classification setting. As discussed in Section 3.2, the optimization in the classification setting is carried out over a matrix of weights $\mathbf{W} \in \mathbb{R}^{p \times c}$. This makes it possible to consider other regularization schemes.

In particular, we apply ideas from *multi-task* learning [32] by viewing each class as a task. More precisely, we use a regularization norm defined by $\Omega_{\text{multi-task}}(\mathbf{W}) \triangleq \sum_{j=1}^p \|\mathbf{W}_j\|$, where $\|\mathbf{W}_j\|$ denotes either the ℓ_2 - or ℓ_∞ -norm of the j -th row of \mathbf{W} . The rationale for the definition of $\Omega_{\text{multi-task}}$ is to assume that the set of relevant voxels is the same across the c different classes, so that sparsity is induced simultaneously over the columns of \mathbf{W} . As a remark, in the ‘‘one-versus-all’’ setting, although the loss functions for the c classes are decoupled, the use of $\Omega_{\text{multi-task}}$ induces a relationship that ties them together.

Note that the tree-structured regularization Ω we consider does not impose a joint pattern-selection across the c different classes. Although a multi-task extension of Ω with ℓ_∞ -norms has recently been proposed [27], the cost of the corresponding proximal operator is significantly higher, which is likely to raise some computational issues in our large-scale experiments.

5.4. Results. We present result of the comparison of our approach based on the hierarchical sparsity-inducing norm (2.2) with the models presented in the previous section. For each method, we computed the cross-validated prediction accuracy and the percentage of non-zero coefficients, *i.e.*, the level of sparsity of the models.

5.4.1. Regression results. The results for the inter-subject regression analysis are given in Table 5.1. The lowest error in prediction accuracy is obtained by the proposed hierarchical structured sparsity approach (Tree ℓ_2 with $\rho = 1$), that also yields one of the lowest (along with greedy) standard deviation indicating that the results are most stable. This can be explained by the fact that the use of local signal averages in the proposed algorithm is a good way to get some robustness to inter-subject variability. We also notice that the sparsity-inducing approaches (Lasso and reweighted ℓ_1) have the highest error in prediction accuracy, probably because the obtained solutions are too sparse, and suffer from the absence of perfect voxel-to-voxel correspondences between subjects.

In terms of sparsity, we can see, as expected, that ridge regression does not yield any sparsity and that the Lasso solution is very sparse (in the feature space, with approximately 7×10^4 voxels). Our method yields a median value of 9.36% of non-zero coefficients (in the augmented space of features, with about 1.4×10^5 nodes in the tree). The maps of weights obtained with Lasso and the hierarchical regularization for one fold, are given in Fig. 5.2. The Lasso yields a scattered and overly sparse pattern of voxels, that is not easily readable, while our approach extracts a pattern of voxels with a compact structure, that clearly outlines brain regions expected to activate differentially for stimuli with different low-level visual properties, *e.g.*, sizes; the early visual cortex in the occipital lobe at the back of the brain. Interestingly, the patterns of voxels show some symmetry between left and right hemispheres, especially in the primary visual cortex which is located at the back and center of the brain. Such an observation matches very well with existing neurosciences knowledge of this brain region that processes the visual contents of both visual hemifields. The weights obtained at different depth level in the tree, corresponding to different scales, show that the largest coefficients are concentrated at the higher scales (scale 6 in Fig. 5.2), showing that the object size

⁵Formally, the depth of the feature j is equal to $\text{depth}(g_j)$, where g_j is the smallest group in \mathcal{G} that contains j (*smallest* is understood here in the sense of the inclusion).

Loss function:	Square		
	Error (mean,std)	P-value w.r.t. Tree ℓ_2 ($\rho = 1$)	Median fraction of non-zeros (%)
Regularization:			
ℓ_2 (Ridge)	(8.3, 4.6)	0.096	100.00
ℓ_1	(12.1, 6.6)	0.013*	0.11
Reweighted ℓ_1	(11.3, 8.8)	0.052	0.10
ℓ_1 (tree weights)	(8.3, 4.7)	0.032*	0.02
Tree ℓ_2 ($\rho = 0.5$)	(7.8, 4.4)	0.137	99.99
Tree ℓ_2 ($\rho = 1$)	(7.1, 4.0)	-	9.36
Tree ℓ_2 ($\rho = 1.5$)	(8.1, 4.2)	0.080	0.04
Tree ℓ_∞ ($\rho = 0.5$)	(8.1, 4.7)	0.080	99.99
Tree ℓ_∞ ($\rho = 1$)	(7.7, 4.1)	0.137	1.22
Tree ℓ_∞ ($\rho = 1.5$)	(7.8,4.1)	0.096	0.04
	Error (mean,std)	P-value w.r.t. Tree ℓ_2 ($\rho = 1$)	Median fraction of non-zeros (%)
Greedy	(7.2, 3.3)	0.5	0.01

TABLE 5.1

Prediction results obtained on fMRI data (see text) for the regression setting. From the left, the first column contains the mean and standard deviation of the test error (unexplained variance), computed over leave-one-subject-out folds. The best performance is obtained with the hierarchical ℓ_2 penalization ($\rho = 1$) constructed from the Ward tree. Statistical significance is assessed with a Wilcoxon two-sample paired signed rank test. The superscript * indicates a rejection at 5%.

cannot be well decoded at the voxel level but requires features formed by more macroscopic clusters of voxels.

5.4.2. Classification results. The results for the inter-subject classification analysis are given in Table 5.2. The best performance is obtained with a multinomial logistic loss function, also using the hierarchical ℓ_2 penalization ($\rho = 1$).

For both ℓ_1 and hierarchical regularizations, one of the three vectors of coefficients obtained for one fold are presented in Fig. 5.3. While for ℓ_1 , the active voxels are scattered all over the brain, the tree ℓ_2 regularization yields clearly delineated sparsity patterns located in the visual areas of the brain. Like for the regression results, the highest coefficients are obtained at scale 6 showing how spatially extended is the brain region involved in the cognitive task. The symmetry of the pattern at this scale is also particularly striking in the primary visual areas. It also extends more anteriorly into the inferior temporal cortex, known for high-level visual processing.

6. Conclusion. In this article, we introduced a hierarchically structured regularization, which takes into account the spatial and multi-scale structure of fMRI data. This approach copes with inter-subject variability in a similar way as feature agglomeration, by averaging neighboring voxels. Although alternative agglomeration strategies do exist, we simply used the criterion which appears as the most natural, Ward’s clustering, and which builds parcels with little variance.

Results on a real dataset show that the proposed algorithm is a promising tool for mining fMRI data. It yields higher prediction accuracy than reference methods, and the map of weights it obtains exhibit a cluster-like structure. It makes them easily readable compared to the overly sparse patterns found by classical sparsity-promoting approaches.

For the regression problem, both the greedy method from [29] and the proposed algorithm yield better results than unstructured and non-hierarchical regularizations. However, in both regression and classification settings, the convex formulation introduced here leads to the best performance while enjoying the guarantees of convex optimization. In particular,

while the greedy algorithm relies on a two-step approach that may be far from optimal, the hierarchical regularization induces simultaneously the selection of the optimal parcellation and the construction of the optimal predictive model, given the initial hierarchical clustering of the voxels. Moreover, convex methods yield predictors that are essentially stable with respect to perturbations of the design or the initial clustering, which is typically not the case of greedy methods.

Finally, it should be mentioned that the performance achieved by this approach in inter-subject problems suggests that it could potentially be used successfully in medical diagnosis problems, where brain images –not necessarily functional images– are used to classify individuals into diseased or control population. Indeed, for difficult problems of that sort, where the reliability of the diagnostic is essential, the stability of models obtained from convex formulations and the interpretability of sparse and localized solutions are useful properties to have in order to provide a credible diagnostic.

Acknowledgments. The authors acknowledge support from the ANR grant ViMAG-INE ANR-08-BLAN-0250-02. The project was also partially supported by a grant from the European Research Council (SIERRA Project).

REFERENCES

- [1] R. G. BARANIUK, V. CEVHER, M. F. DUARTE, AND C. HEGDE, *Model-based compressive sensing*, IEEE Transactions on Information Theory, 56 (2010), pp. 1982–2001.
- [2] A. BECK AND M. TEOULLE, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM Journal on Imaging Sciences, 2 (2009), pp. 183–202.
- [3] E. J. CANDÈS, M. B. WAKIN, AND S. P. BOYD, *Enhancing sparsity by reweighted l_1 minimization*, Journal of Fourier Analysis and Applications, 14 (2008), pp. 877–905.
- [4] M. K. CARROLL, G. A. CECCHI, I. RISH, R. GARG, AND A. R. RAO, *Prediction and interpretation of distributed neural activity with sparse models*, NeuroImage, 44 (2009), pp. 112 – 122.
- [5] S. S. CHEN, D. L. DONOHO, AND M. A. SAUNDERS, *Atomic decomposition by basis pursuit*, SIAM Journal on Scientific Computing, 20 (1998), pp. 33–61.
- [6] D. B. CHKLOVSKII AND A. A. KOULAKOV, *Maps in the brain: What can we learn from them?*, Annual Review of Neuroscience, 27 (2004), pp. 369–392.
- [7] P. L. COMBETTES AND J.-C. PESQUET, *Proximal splitting methods in signal processing*, in Fixed-Point Algorithms for Inverse Problems in Science and Engineering, Springer, 2010.
- [8] D. D. COX AND R. L. SAVOY, *Functional magnetic resonance imaging (fMRI) "brain reading": detecting and classifying distributed patterns of fMRI activity in human visual cortex*, NeuroImage, 19 (2003), pp. 261–270.
- [9] S. DEHAENE, G. LE CLEC'H, L. COHEN, J.-B. POLINE, P.-F. VAN DE MOORTELE, AND D. LE BIHAN, *Inferring behavior from functional brain images*, Nature Neuroscience, 1 (1998), p. 549.
- [10] D. L. DONOHO AND I. M. JOHNSTONE, *Adapting to unknown smoothness via wavelet shrinkage.*, Journal of the American Statistical Association, 90 (1995).
- [11] E. EGER, C. KELL, AND A. KLEINSCHMIDT, *Graded size sensitivity of object exemplar evoked activity patterns in human loc subregions*, J. Neurophysiol., 100(4):2038–47 (2008).
- [12] G. FLANDIN, F. KHERIF, X. PENNEC, G. MALANDAIN, N. AYACHE, AND J.-B. POLINE, *Improved detection sensitivity in functional MRI data using a brain parcelling technique*, in Medical Image Computing and Computer-Assisted Intervention (MICCAI'02), 2002, pp. 467–474.
- [13] K. J. FRISTON, A. P. HOLMES, K. J. WORSLEY, J. B. POLINE, C. FRITH, AND R. S. J. FRACKOWIAK, *Statistical parametric maps in functional imaging: A general linear approach*, Human Brain Mapping, 2 (1995), pp. 189–210.
- [14] L. GROSENICK, S. GREER, AND B. KNUTSON, *Interpretable classifiers for FMRI improve prediction of purchases*, IEEE Transactions on Neural Systems and Rehabilitation Engineering, 16 (2009), pp. 539–548.
- [15] I. GUYON, J. WESTON, S. BARNHILL, AND V. VAPNIK, *Gene selection for cancer classification using support vector machines*, Machine Learning, 46 (2002), pp. 389–422.
- [16] T. HASTIE, R. TIBSHIRANI, AND J. FRIEDMAN, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*, Springer, 2009.
- [17] J. HUANG, T. ZHANG, AND D. METAXAS, *Learning with structured sparsity*, in Proceedings of the International Conference on Machine Learning (ICML), 2009.

- [18] L. JACOB, G. OBOZINSKI, AND J.-P. VERT, *Group Lasso with overlaps and Graph Lasso*, in Proceedings of the International Conference on Machine Learning (ICML), 2009.
- [19] R. JENATTON, J.-Y. AUDIBERT, AND F. BACH, *Structured variable selection with sparsity-inducing norms*, tech. report, Preprint arXiv:0904.3523, 2009.
- [20] R. JENATTON, A. GRAMFORT, V. MICHEL, G. OBOZINSKI, F. BACH, AND B. THIRION, *Multi-scale mining of fMRI data with hierarchical structured sparsity*, in International Workshop on Pattern Recognition in Neuroimaging (PRNI), 2011.
- [21] R. JENATTON, J. MAIRAL, G. OBOZINSKI, AND F. BACH, *Proximal methods for hierarchical sparse coding*, tech. report, Preprint arXiv:1009.2139v2, 2010. Submitted to the Journal of Machine Learning Research.
- [22] ———, *Proximal methods for sparse hierarchical dictionary learning*, in Proceedings of the International Conference on Machine Learning (ICML), 2010.
- [23] S. C. JOHNSON, *Hierarchical clustering schemes*, Psychometrika, 32 (1967), pp. 241–254.
- [24] S. KIM AND E. P. XING, *Tree-guided group Lasso for multi-task regression with structured sparsity*, in Proceedings of the International Conference on Machine Learning (ICML), 2010.
- [25] S. LACONTE, S. STROTHER, V. CHERKASSKY, J. ANDERSON, AND X. HU, *Support vector machines for temporal classification of block design fMRI data*, NeuroImage, 26 (2005), pp. 317 – 329.
- [26] E. L. LEHMANN AND J. P. ROMANO, *Testing statistical hypotheses*, Springer Verlag, 2005.
- [27] J. MAIRAL, R. JENATTON, G. OBOZINSKI, AND F. BACH, *Network flow algorithms for structured sparsity*, in Advances in Neural Information Processing Systems, 2010.
- [28] F. DE MARTINO, G. VALENTE, N. STAEREN, J. ASHBURNER, R. GOEBEL, AND E. FORMISANO, *Combining multivariate voxel selection and support vector machines for mapping and classification of fMRI spatial patterns*, NeuroImage, 43 (2008), pp. 44 – 58.
- [29] V. MICHEL, E. EGER, C. KERIBIN, J.-B. POLINE, AND B. THIRION, *A supervised clustering approach for extracting predictive information from brain activation images*, MMBIA'10, (2010).
- [30] V. MICHEL, A. GRAMFORT, G. VAROQUAUX, E. EGER, AND B. THIRION, *Total variation regularization for fMRI-based prediction of behaviour*, Medical Imaging, IEEE Transactions on, PP (2011), p. 1.
- [31] Y. NESTEROV, *Gradient methods for minimizing composite objective function*, tech. report, Center for Operations Research and Econometrics (CORE), Catholic University of Louvain, 2007.
- [32] G. OBOZINSKI, B. TASKAR, AND M.I. JORDAN, *Joint covariate selection and joint subspace selection for multiple classification problems*, Statistics and Computing, 20 (2010), pp. 231–252.
- [33] P. RAMACHANDRAN AND G. VAROQUAUX, *Mayavi: 3d visualization of scientific data*, Computing in Science Engineering, 13 (2011), pp. 40 –51.
- [34] R. RIFKIN AND A. KLAUTAU, *In defense of one-vs-all classification*, Journal of Machine Learning Research, 5 (2004), pp. 101–141.
- [35] J. RISSMAN, H. T. GREELY, AND A. D. WAGNER, *Detecting individual memories through the neural decoding of memory states and past experience*, Proceedings of the National Academy of Sciences, 107 (2010), pp. 9849–9854.
- [36] S. RYALI, K. SUPEKAR, D. A. ABRAMS, AND V. MENON, *Sparse logistic regression for whole-brain classification of fMRI data*, NeuroImage, 51 (2010), pp. 752 – 764.
- [37] B. SCHÖLKOPF AND A. J. SMOLA, *Learning with kernels: support vector machines, regularization, optimization, and beyond*, MIT Press, 2002.
- [38] B. THIRION, G. FLANDIN, P. PINEL, A. ROCHE, P. CIUCIU, AND J.-B. POLINE, *Dealing with the shortcomings of spatial normalization: Multi-subject parcellation of fMRI datasets*, Hum. Brain Mapp., 27 (2006), pp. 678–693.
- [39] R. TIBSHIRANI, *Regression shrinkage and selection via the Lasso*, Journal of the Royal Statistical Society. Series B, (1996), pp. 267–288.
- [40] K. UGURBIL, L. TOTH, AND D. KIM, *How accurate is magnetic resonance imaging of brain function?*, Trends in Neurosciences, 26 (2003), pp. 108 – 114.
- [41] J. H. WARD, *Hierarchical grouping to optimize an objective function*, Journal of the American Statistical Association, 58 (1963), pp. 236–244.
- [42] S. J. WRIGHT, R. D. NOWAK, AND M. A. T. FIGUEIREDO, *Sparse reconstruction by separable approximation*, IEEE Transactions on Signal Processing, 57 (2009), pp. 2479–2493.
- [43] O. YAMASHITA, M. SATO, T. YOSHIOKA, F. TONG, AND Y. KAMITANI, *Sparse estimation automatically selects voxels relevant for the decoding of fMRI activity patterns*, NeuroImage, 42 (2008), pp. 1414 – 1429.
- [44] P. ZHAO, G. ROCHA, AND B. YU, *The composite absolute penalties family for grouped and hierarchical variable selection*, Annals of Statistics, 37 (2009), pp. 3468–3497.
- [45] H. ZOU AND T. HASTIE, *Regularization and variable selection via the elastic net*, Journal of the Royal Statistical Society. Series B, 67 (2005), pp. 301–320.

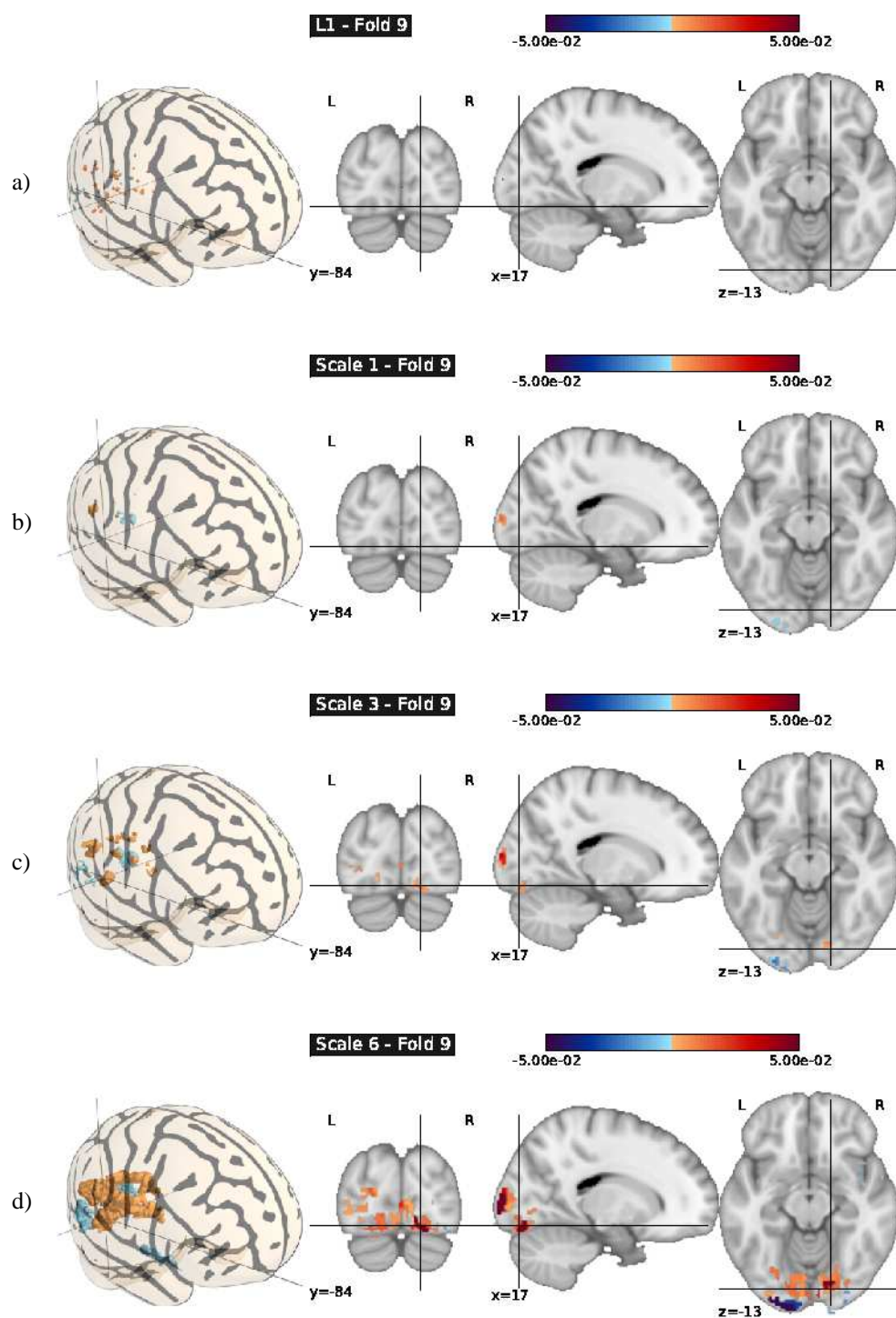


FIG. 5.2. Maps of weights obtained using different regularizations in the regression setting. (a) ℓ_1 regularization - We can notice that the predictive pattern obtained is excessively sparse, and is not easily readable despite being mainly located in the occipital cortex. (b-d) tree ℓ_2 regularization ($\rho = 1$) at different scales - In this case, the regularization algorithm extracts a pattern of voxels with a compact structure, that clearly outlines early visual cortex which is expected to discriminate between stimuli of different sizes. 3D images were generated with Mayavi [33].

Loss function:	Square ("one-versus-all")		
	Error (mean,std)	P-value w.r.t. Tree ℓ_2 ($\rho = 1$)-ML	Median fraction of non-zeros (%)
Regularization:			
ℓ_2 (Ridge)	(29.2, 5.9)	0.004*	100.00
ℓ_1	(33.3, 6.8)	0.004*	0.10
ℓ_1/ℓ_2 (Multi-task)	(31.7, 9.5)	0.004*	0.12
ℓ_1/ℓ_∞ (Multi-task)	(33.3,13.6)	0.009*	0.22
Tree ℓ_2 ($\rho = 0.5$)	(25.8, 9.2)	0.004*	99.93
Tree ℓ_2 ($\rho = 1$)	(25.0, 5.5)	0.027*	10.08
Tree ℓ_2 ($\rho = 1.5$)	(24.2, 9.9)	0.130	0.05
Tree ℓ_∞ ($\rho = 0.5$)	(30.8, 8.8)	0.004*	59.49
Tree ℓ_∞ ($\rho = 1$)	(24.2, 7.3)	0.058	1.21
Tree ℓ_∞ ($\rho = 1.5$)	(25.8, 10.7)	0.070	0.04
Loss function:	Logistic ("one-versus-all")		
	Error (mean,std)	P-value w.r.t. Tree ℓ_2 ($\rho = 1$)-ML	Median fraction of non-zeros (%)
Regularization:			
ℓ_2 (Ridge)	(25.0, 9.6)	0.008*	100.00
ℓ_1	(34.2, 15.9)	0.004*	0.55
ℓ_1/ℓ_2 (Multi-task)	(31.7, 8.6)	0.002*	47.35
ℓ_1/ℓ_∞ (Multi-task)	(33.3, 10.4)	0.002*	99.95
Tree ℓ_2 ($\rho = 0.5$)	(25.0, 9.6)	0.007*	99.93
Tree ℓ_2 ($\rho = 1$)	(20.0, 11.2)	0.250	7.88
Tree ℓ_2 ($\rho = 1.5$)	(18.3, 6.6)	0.500	0.06
Tree ℓ_∞ ($\rho = 0.5$)	(30.8, 10.4)	0.004*	59.42
Tree ℓ_∞ ($\rho = 1$)	(24.2, 6.1)	0.035*	0.60
Tree ℓ_∞ ($\rho = 1.5$)	(21.7, 8.9)	0.125	0.03
Loss function:	Multinomial logistic (ML)		
	Error (mean,std)	P-value w.r.t. Tree ℓ_2 ($\rho = 1$)-ML	Median fraction of non-zeros (%)
Regularization:			
ℓ_2 (Ridge)	(24.2, 9.2)	0.035*	100.00
ℓ_1	(25.8, 12.0)	0.004*	97.95
ℓ_1/ℓ_2 (Multi-task)	(26.7, 7.6)	0.007*	30.24
ℓ_1/ℓ_∞ (Multi-task)	(26.7, 11.6)	0.002*	99.98
Tree ℓ_2 ($\rho = 0.5$)	(22.5, 8.8)	0.070	83.06
Tree ℓ_2 ($\rho = 1$)	(16.7, 10.4)	-	4.87
Tree ℓ_2 ($\rho = 1.5$)	(18.3, 10.9)	0.445	0.02
Tree ℓ_∞ ($\rho = 0.5$)	(26.7, 11.6)	0.015*	48.82
Tree ℓ_∞ ($\rho = 1$)	(22.5, 13.0)	0.156	0.34
Tree ℓ_∞ ($\rho = 1.5$)	(21.7, 8.9)	0.460	0.05
	Error (mean,std)	P-value w.r.t. Tree ℓ_2 ($\rho = 1$)-ML	Median fraction of non-zeros (%)
Greedy	(21.6, 14.5)	0.001*	0.01

TABLE 5.2

Prediction results obtained on fMRI data (see text) for the multi-class classification setting. From the left, the first column contains the mean and standard deviation of the test error (percentage of misclassification), computed over leave-one-subject-out folds. The best performance is obtained with the hierarchical ℓ_2 penalization ($\rho = 1$) constructed from the Ward tree, coupled with the multinomial logistic loss function. Statistical significance is assessed with a Wilcoxon two-sample paired signed rank test. The superscript * indicates a rejection at 5%.

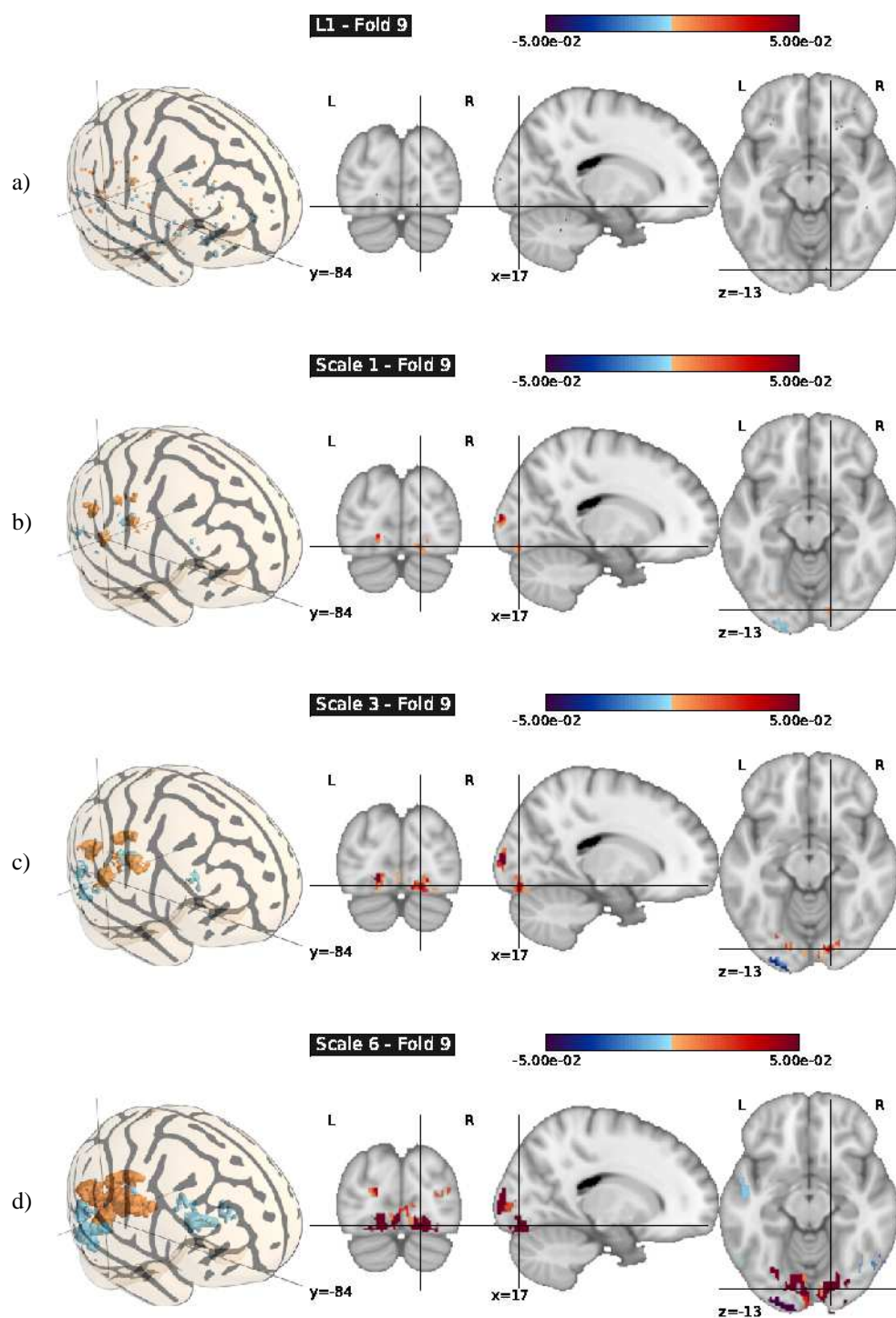


FIG. 5.3. Maps of weights obtained using different regularizations in the classification setting. (a) ℓ_1 regularization - We can notice that the predictive pattern obtained is excessively sparse, and is not easily readable with voxels scattered all over the brain. (b-d) tree regularization at different scales - In this case, the regularization algorithm extracts a pattern of voxels with a compact structure, that clearly outlines early visual cortex which is expected to discriminate between stimuli of different sizes.