Published in final edited form as:

Neuroimage. 2018 July 15; 175: 230–245. doi:10.1016/j.neuroimage.2018.03.040.

SMAC: Spatial multi-category angle-based classifier for high-dimensional neuroimaging data[★]

Leo Yu-Feng Liu^a, Yufeng Liu^{a,b,c}, Hongtu Zhu^{b,d,*}, and for the Alzheimer's Disease Neuroimaging Initiative

^aDepartment of Statistics and Operations Research, The University of North Carolina at Chapel Hill, Chapel Hill, NC, USA

^bDepartment of Biostatistics, The University of North Carolina at Chapel Hill, Chapel Hill, NC, USA

^cDepartment of Genetics, Carolina Center for Genome Sciences, Lineberger Comprehensive Cancer Center (LCCC), The University of North Carolina at Chapel Hill, Chapel Hill, NC, USA

^dDepartment of Biostatistics, The University of Texas MD Anderson Cancer Center, Houston, TX, USA

Abstract

With the development of advanced imaging techniques, scientists are interested in identifying imaging biomarkers that are related to different subtypes or transitional stages of various cancers, neuropsychiatric diseases, and neurodegenerative diseases, among many others. In this paper, we propose a novel spatial multi-category angle-based classifier (SMAC) for the efficient identification of such imaging biomarkers. The proposed SMAC not only utilizes the spatial structure of high-dimensional imaging data but also handles both binary and multicategory classification problems. We introduce an efficient algorithm based on an alternative direction method of multipliers to solve the large-scale optimization problem for SMAC. Both our simulation and real data experiments demonstrate the usefulness of SMAC.

Keywords

ADMM; Alzheimer's disease; Angle-based classifier; Fused lasso; Large margin classifier; Neuroimaging classification

Introduction

With advances in modern imaging technology, it is becoming increasingly prevalent to collect high-dimensional imaging data (e.g., magnetic resonance imaging [MRI]) in order to extract imaging biomarkers (or features) that are useful for various tasks, including disease detection, diagnosis, prognosis, and treatment, among many others (Chen et al., 1998; Lopez et al., 2009; Ramírez et al., 2009). For many diseases, such as Alzheimer's disease (AD) and

[★]Corresponding author. Department of Biostatistics, The University of Texas MD Anderson Cancer Center, Houston, TX, USA. hzhu5@mdanderson.org (H. Zhu).

breast cancer, it is expected that medical images contain clinically relevant information associated with their pathophysiology. A critical challenge is determining how to build a predictive model (or classifier) that can classify patients into clinically meaningful subgroups according to their imaging data. Such a model may improve the clinical care of these patients and possibly slow their disease progression.

In the current literature, there exist two groups of classification methods for imaging data, including feature-based analysis and image-based analysis. Feature-based analysis consists of (i) converting medical images into a set of features and (ii) building classifiers based on these extracted features. Standard feature extraction methods often extract some summary statistics (e.g., mean imaging intensity) in either segmented tumors or prefixed regions of interest (ROIs) in a template space. For example, Rusinek et al. (2004) used the partial volumes of the brain and cerebrospinal fluid (CSF) to classify AD versus normal control (NC), and Zhu et al. (2014) built a multi-category classifier using sparse linear discriminant analysis based on features extracted from 93 ROIs of both MRI and positron emission technology (PET) images. More examples of feature-based analysis can be found in Xu et al. (2000), Busatto et al. (2003), Colliot et al. (2008), and Yu et al. (2014). The major drawback of these feature-based methods is that they require knowledge of spatial segmentation to identify meaningful ROIs in order to extract informative, discriminating and independent features for the classification task.

Image-based analysis, however, uses raw imaging data across all grid points. Two key advantages of using raw imaging data include potential gain in classification accuracy and spatially interpretable coefficient maps of the classifiers in the original imaging space. The main challenges for image-based analysis include (i) high dimensionality, (ii) complex spatial information and (iii) noisy functional data. For example, a typical T1-weighted MR image of size $256 \times 256 \times 256$ will yield a 16, 777, 216 dimensional space, and due to the inherent biological structure of the brain, these data also have complex spatial correlation and smoothness.

Many methods in the literature apply a pre-screening procedure to reduce the dimensionality of the imaging data, and build classifiers in the reduced imaging space. For example, Liu et al. (2012) applied the ensemble of multiple classifiers based on randomly selected patches of the MR images, and Hinrichs et al. (2011) built multiple kernel support vector machines based on 2000 to 250,000 features selected by voxel-wise t-tests. The pre-screening procedure can significantly reduce the computational cost in estimating the classifiers, but potentially loses important predictive information. On the other hand, many regularization techniques have been proposed to directly handle high-dimensional data, including imaging data as a special case (Grosenick et al., 2008, 2009; Yamashita et al., 2008; Van Gerven and Heskes, 2012). For instance, Yamashita et al. (2008) proposed a method by imposing L_2 norm regularization to logistic regression for classification of functional MRI data in various tasks; whereas Casanova et al. (2011) applied elastic-net penalized regression to distinguish between patients with AD versus NCs based on both gray matter and white matter segmentation maps. These regularization methods perform simultaneous estimation of coefficients across all voxels and select the predictive voxels. Since most standard regularization methods do not account for the spatial structure of imaging data, their

resulting classifiers usually contain only isolated voxels; thus, it can be difficult to interpret the results. Moreover, standard sparsity penalties, such as L_1 , can be sub-optimal for the high-dimensional prediction problems considered here, since the effect of high-dimensional imaging data on certain categories is often spatially clustered and non-sparse.

To effectively handle imaging data, it is critically important to utilize the spatial smoothness and correlation of imaging data in the construction of classifiers. For instance, Grosenick et al. (2013) proposed a spatial smoothing classifier based on the GraphNet penalty. Furthermore, Watanabe et al. (2014) developed a spatial support vector machine (SSVM) classifier based on the fused lasso (FL) and GraphNet penalties. These methods yield meaningful coefficient images and achieve good accuracy for binary neuroimage classification, but are not directly applicable to multi-category classification problems.

The aim of this paper is to develop a spatial multi-category angle-based classifier (SMAC) for high-dimensional imaging data. Compared with the existing methods in the literature, three major methodological contributions of this paper are as follows:

- The proposed SMAC not only utilizes the spatial structure of images, but also extends the angle-based classification framework recently developed by Zhang and Liu (2014) to perform simultaneous multi-category classification of imaging data.
- We use a hybrid of a generalized total variation (TV) penalty (Tibshirani et al., 2005) and a sparse L_1 penalty, namely an FL penalty, to identify spatially aggregated clusters that are important for discriminating different classes. Our methods are able to deliver competitive classification accuracy and interpretable imaging biomarkers.
- We have developed the SMAC package by using both MATLAB and Python. The MATLAB codes have been released through the websites “<https://www.nitrc.org/projects/smac>” and “<https://github.com/BIG-S2/SMAC>”. Our package includes a graphical user interface that is freely downloadable from the same websites. Our SMAC package can handle 1-dimensional (1-D) curves, 2-dimensional (2-D) surfaces, and 3-dimensional (3-D) volumes.

The rest of the paper is organized as follows. In Section 2, we introduce the SMAC framework and describe an optimization algorithm to efficiently estimate the model coefficients. We use two simulation experiments and the Alzheimer's Disease Neuroimaging Initiative (ADNI) data in Section 3 to examine the finite-sample performance of SMAC. In Section 4, we conclude with some discussion.

Methods and materials

Data structure

One important classification problem in the neuroimaging literature is to predict the disease status of patients based on their neurological images. The class label is denoted by a categorical response variable y , usually taking values of 1, 2, ..., K , indicating K different

classes of interest. The covariate $\mathbf{X} = \{x_d; d \in \mathcal{D}\} \in \mathbb{R}^p$ represents the observed imaging data, where $\mathcal{D}$ denotes the spatial structure of the image, which can be a 1-D curve, 2-D surface or 3-D volume, and d is a vector of length 1, 2 or 3, indicating the location of the underlying voxel in the image. Without loss of generality, we focus on 3-D real valued images in this paper, and use p as the dimension of the imaging data, which equals the total number of voxels in the image.

We use bold symbols to represent the imaging variables, such as $\mathbf{X}$ and use regular symbols for scalar and other non-image variables, such as y . The notation $[n]$ represents the set $\{1, 2, \dots, n\}$.

Statistical classification framework

For a K -category classification problem, a statistical classifier builds a map from the covariate space $\mathbb{R}^p$ to the category space $\{1, \dots, K\}$. Given a new imaging observation $\mathbf{X}^*$, the classifier predicts the associated class label y^* as $\hat{y}^*$. To build the classifier, many statistical procedures can be fitted into the regularization framework of *loss + penalty*. A loss function $\ell(\cdot)$ is introduced to ensure the goodness of fit of the resulting model to the training data. Two groups of loss functions that are commonly used in the literature include likelihood-based and margin-based loss functions. Likelihood-based methods usually impose some assumption of probability distributions on the data and then establish the classification rule by solving some parametric statistical models. Examples of these methods include Fishers linear discriminant analysis (LDA) (Fisher, 1936) and logistic regression (Hastie et al., 2005). In contrast, margin-based methods solve the classification problems without imposing a strong distributional assumption on the data. Specifically, a margin-based method uses a functional margin as the input of the loss function $\ell(\cdot)$. The values of the functional margins are directly associated with the accuracy of the class label assignment. For binary classification with the class label $W_y \in \{\pm 1\}$ for $y \in \{1, 2\}$, one can obtain a function $f(x)$ and use $\hat{W}_y = \text{sign}(f(\mathbf{X}))$ as a classification rule. In this case, the functional margin is defined as $W_y f(\mathbf{X})$, indicating the correctness of the classification. Our proposed classifier belongs to margin-based methods.

When dealing with high-dimensional data, a regularization term is usually added to the loss function to prevent the models from over-fitting the training data. The choice of the regularization term is based on prior knowledge of the data structure and the properties of the specific penalty. For instance, the L_1 norm penalty can be utilized to learn the sparse structure of data (Tibshirani, 1996), and the L_2 type of penalties encourage continuous shrinkage in the estimation (Zou and Hastie, 2005). To choose the penalty term for handling the neuroimaging data, it is necessary to account for its high dimensionality and complex imaging structure. A desired penalty should encourage sparsity, while incorporating the spatial structure of the imaging data.

Binary large-margin classifiers

Many “off the shelf” classifiers are potential candidates for neuroimaging classification. Examples range from the very classical LDA (Fisher, 1936) and logistic regression (Hastie

et al., 2005) to the recent machine learning techniques, such as the support vector machine (Boser et al., 1992) and boosting (Friedman et al., 2000). The choice of the classifier depends on the data structure and the goal of classification. However, there is no clear guideline about which classifier to choose in each complicated case. Liu et al. (2011) proposed a large-margin unified classifier (LUM), covering a rich family of classification methods, which allows us to tune our loss function within the rich LUM family to obtain a satisfactory solution. In this paper, we choose a special LUM loss function which has the following form:

$$l(u) = \begin{cases} 1 - u, & \text{if } u < 0; \\ e^{-u}, & \text{if } u \geq 0. \end{cases} \quad (1)$$

This special loss can be viewed as a hybrid of the support vector machine and AdaBoost, which allows us to maximize the separation margin and dynamically assign weights in “weak” learners (Freund and Schapire, 1997). We refer readers to the original paper for further details of the LUM loss.

Despite the potential improvement in classification performance when using LUM, this classifier was originally proposed to solve binary classification problems. The extension to multi-category cases requires additional effort. We address this issue in the following section.

Multi-category large-margin classifiers

To handle multi-category data, one simple approach is to conduct binary classification sequentially via the one-versus-one or one-versus-the-rest scheme in order to predict the class labels. These methods have been proven to be suboptimal when there is no dominating class (Liu and Yuan, 2011). Other classifiers solve the classification problem simultaneously by mapping covariates to a vector with the length equal to the total number of categories. Such classifiers can be found in Zhu and Hastie (2005), Zhu et al. (2009) and Liu and Yuan (2011). A sum-to-zero constraint on the predicted vector is usually applied to achieve desirable theoretical properties, but may increase the complexity of the corresponding optimization. Without this constraint, Zhang and Liu (2014) proposed a multi-category angle-based classifier (MAC) that can achieve the Fisher consistency and some other desirable properties.

For a K -category classification problem ($K \geq 2$), MAC creates a map from the class labels $y \in [K]$ to the vertices of a regular simplex in the $(K-1)$ -dimensional space, i.e.,

$$W_y = \begin{cases} (K-1)^{-1/2} e, & \text{if } y = 1; \\ \frac{1 + K^{1/2}}{(K-1)^{3/2}} e + \left(\frac{K}{K-1}\right)^{1/2} e_{y-1}, & \text{if } y \in [K] / 1, \end{cases} \quad (2)$$

where $\xi \in \mathbb{R}^{K-1}$ is a vector with all elements being 1, and $e_y \in \mathbb{R}^{K-1}$ is a vector such that all elements are 0, except that the y -th component is 1. Note that for $K=2$, it reduces to the traditional binary classification with labels $W_y \in \{\pm 1\}$. Due to the property of the regular simplexes, the angles between any two projected class labels are equal, i.e., $\angle(W_y, W_{y'}) = C_K$ for all $y \neq y'$.

Instead of directly using the original class label y , MAC uses the projected class label W_y to solve the multi-category problem. In particular, we construct a function that maps the covariate X to the same $K-1$ dimensional space, i.e., $f: \mathbb{R}^p \rightarrow \mathbb{R}^{K-1}$, and use the angle between $f(X)$ and W_y for $y \in [K]$ to determine the prediction rule, i.e.,

$$\hat{y} = \arg \min_y \angle(W_y, f(X)).$$

According to the “law of cosine”, this is equivalent to

$$\hat{y} = \arg \max_y \langle W_y, f(X) \rangle, \quad (3)$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product of two vectors. The inner product essentially plays the role of the functional margin in MAC, and the empirical risk minimization (ERM) is, therefore defined as follows:

$$\min_{f \in \mathcal{F}} \left\{ \sum_{i=1}^n l(\langle W_{y_i}, f(X_i) \rangle) + \lambda J(f) \right\}, \quad (4)$$

where $l(\cdot)$ is the margin-based loss function defined by equation (1) and $J(f)$ is the penalty term with the tuning parameter λ , which controls the strength of regularization.

Considering the specialty of voxel-based neuroimaging classification, we narrow the function space $\mathcal{F}$ to linear functions, so that the coefficients of $f(\cdot)$ are voxel-wisely matched with the structure of the imaging covariate X , i.e.,

$$\begin{aligned} f(X) &= (f_1(X), f_2(X), \dots, f_{K-1}(X))^T, \\ f_j(X) &= \beta_{j,0} + x_1 \beta_{j,1} + \dots + x_p \beta_{j,p} \text{ for } j \in [K-1]. \end{aligned} \quad (5)$$

Notice that $\beta_j = (\beta_{j,1}, \dots, \beta_{j,p})^T$ has a one-to-one correspondence with the imaging data $X = (x_1, \dots, x_p)^T$. Thus, it can be also defined in the original imaging space of the covariates. In this case, we denote β_j as the coefficient image of the fitted classifier.

For a K -category classification problem, we have $K - 1$ coefficient images. In order to match the coefficient images with the K class labels, we denote the reconstructed coefficient images β_y^* , $y \in [K]$ of the same dimension of β_j as follows,

$$\beta_y^* = \sum_{j=1}^{K-1} W_{y,j} \bullet \beta_j, \quad (6)$$

where $W_{y,j}$ is the j -th element of the project class label W_y in Equation (2) and “ $\bullet$ ” denotes the element-wise product.

Note that β_y^* has the one-to-one correspondence with the class label y . Additionally, since $\sum_{y=1}^K W_y = 0$ according to Equation (2), we have the sum-to-zero constraint on β_y^* 's as well, i.e., $\sum_{k=1}^K \beta_y^* = 0$. These properties ensure that the reconstructed coefficient images are comparable with the coefficient images obtained from other linear classification models with the sum-to-zero constraint, such as logistic regression.

Spatial smoothing regularization

The penalty term $\mathcal{J}(f)$ in problem (4) not only plays an important role of preventing the resulting classifier from over-fitting, but also helps to achieve some desired structure in the coefficient images. For imaging classification, unpenalized estimation often yields dense coefficients, but requires additional thresholding (or feature selection) to identify meaningful biomarkers. In contrast, the use of sparse penalties alone, such as lasso and the elastic net, leads to coefficient images with isolated voxels, which can be difficult to interpret. The use of spatial smoothing penalties not only captures the spatial smoothness in the image space, but also yields biologically interpretable coefficient images. For instance, Grosenick et al. (2013) proposed a spatial smoothing penalty, GraphNet, that incorporates the spatial structure in the elastic net penalization. However, the GraphNet penalty yields global smoothness in coefficient images, so it may be suboptimal in preserving sharp edges.

We introduce the generalized FL penalty (Tibshirani, 2011) to capture the spatial structure of imaging data. For an image $I = \{I(d) \in \mathbb{R} : d \in \mathcal{D}\}$, the discrete imaging intensities are evaluated at grid points $d = (d_1, d_2, d_3)^T \in \mathbb{R}^3$ in a compact set $\mathcal{D}$. The FL penalty is a weighted mixture of the L_1 and TV penalty on the imaging intensities. The L_1 penalty encourages both shrinkage and sparseness (Tibshirani, 1996); whereas the TV penalty regularizes the differences between the consecutive elements in the estimation. We denote the latter as the TV-I penalty. Its discrete formulation is defined as follows:

$$\text{TV-I}(I) = \sum_{d_1=1}^{D_1} \sum_{d_2=1}^{D_2} \sum_{d_3=1}^{D_3} \left\| \nabla I_{d_1, d_2, d_3} \right\|, \quad (7)$$

where $\|\cdot\|_1$ denotes the L_1 norm, D_1 , D_2 and D_3 respectively represent the total number of voxels along each dimension, and ∇ is the discrete differential operator such that $\nabla \mathbf{I}_{d_1, d_2, d_3} = (\nabla_1 \mathbf{I}_{d_1, d_2, d_3}, \nabla_2 \mathbf{I}_{d_1, d_2, d_3}, \nabla_3 \mathbf{I}_{d_1, d_2, d_3})^T$. Moreover, $\nabla_1 \mathbf{I}_{d_1, d_2, d_3}$ is defined as

$$\nabla_1 \mathbf{I}_{d_1, d_2, d_3} = \begin{cases} \mathbf{I}_{d_1, d_2, d_3} - \mathbf{I}_{d_1+1, d_2, d_3} & \text{if } 1 \leq d_1 \leq D_1 - 1, \\ 0 & \text{if } d_1 = D_1, \end{cases}$$

and $\nabla_2 \mathbf{I}_{d_1, d_2, d_3}$ and $\nabla_3 \mathbf{I}_{d_1, d_2, d_3}$ can be similarly defined.

The TV-I penalty penalizes the discrete gradient of the imaging function $\mathbf{I}(\cdot)$. It encourages the spatial smoothness of $\mathbf{I}(\cdot)$, while capturing its sharp edges. This property allows us to efficiently detect important blobs. However, in some cases, the TV-I penalty tends to yield images with block-wise constant blobs (Rudin et al., 1992), which might erase too many details. For this reason, we introduce the second-order TV penalty, denoted TV-II, which can capture blobs with a continuous change of intensity by imposing the regularization on the Hessian matrix of $\mathbf{I}(\cdot)$, which encourages the gradual fade of $\mathbf{I}(\cdot)$ in the space. The discrete formulation of TV-II is defined as follows:

$$\text{TV-II}(\mathbf{I}) = \sum_{d_1=1}^{D_1-2} \sum_{d_2=1}^{D_2-2} \sum_{d_3=1}^{D_3-2} \left\| H(\mathbf{I}_{d_1, d_2, d_3}) \right\|_1, \quad (8)$$

where $H(\mathbf{I}_{d_1, d_2, d_3}) = (\nabla_m (\nabla_{m'} (\mathbf{I}_{d_1, d_2, d_3})))_{1 \leq m, m' \leq 3}$ and $\|\cdot\|_1$ denotes the entry-wise L_1 norm of a matrix.

Note that the calculation of both gradient and Hessian operators can be represented as matrix multiplication on the vectorized images. In particular, the TV-I in (7) can be represented as

$$\text{TV-I}(\mathbf{I}) = \|\mathbf{D} \times \mathbf{I}\|_1,$$

where $\mathbf{D}$ is the discrete derivative operator that contains the differencing operation along each of the 3 dimensions of the imaging domain.

Similarly, the TV-II penalty can be represented as

$$\text{TV-II}(\mathbf{I}) = \|\mathbf{D}_{II} \mathbf{D} \times \mathbf{I}\|_1,$$

where $\mathbf{D}_{II} = \text{diag}\{\mathbf{D}, \mathbf{D}, \mathbf{D}\}$ is a diagonal block matrix, with 3 copies of matrix $\mathbf{D}$ representing the operations along each dimension.

For problem (4), we have $K-1$ coefficient images for a K category classification problem and can denote $\boldsymbol{\beta} = (\boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_{K-1})^T$ as the vector of all the imaging coefficients, as denoted in equation (5). The associated TV-I penalty is defined as

$$\text{TV-I}(\beta) = \sum_{k=1}^{K-1} \text{TV-I}(\beta_k) = \sum_{k=1}^{K-1} \left\| \mathbf{D} \times \beta_k \right\|_1 = \left\| \mathbf{C}_I \beta \right\|_1,$$

where $\mathbf{C}_I = [\mathbf{D}, \dots, \mathbf{D}]$ is $K-1$ copies of the operator $\mathbf{D}$. Similarly, we can define

$$\text{TV-II}(\beta) = \left\| \mathbf{C}_{II} \beta \right\|_1,$$

where $\mathbf{C}_{II} = [\mathbf{D}_{II}\mathbf{D}, \dots, \mathbf{D}_{II}\mathbf{D}]$ is $K-1$ copies of the matrix $\mathbf{D}_{II}\mathbf{D}$.

Finally, the EMR problem in (4) can be reformulated as follows:

$$\min_{\beta \in \mathbb{R}^{(K-1)(p+1)}} \left\{ \sum_{i=1}^n l\left(\left\langle W_{y_i}, f(X_i) \right\rangle\right) + \text{FL}(\beta) \right\}, \quad (9)$$

where $l(\cdot)$ is the loss function in (1), $f(\cdot)$ is a system of linear functions defined in (5), and $\text{FL}(\beta) = \lambda_1 \|\beta\|_1 + \lambda_2 \|\mathbf{C}\beta\|_1$ is the FL penalty, in which λ_1 and λ_2 are two non-negative tuning parameters and $\mathbf{C} = \mathbf{C}_I$ for TV-I or $\mathbf{C}_{II}$ for TV-II.

Algorithm

The optimization in problem (9) is a mixture of smooth and non-smooth convex optimization. Many iterative proximal algorithms can be adopted here to solve this problem, such as ISTA and FISTA (Beck and Teboulle, 2009). However, the evaluation of the Lipschitz constant and the proximal operators can be computationally expensive in this case. Instead, we introduce an alternative direction method of multipliers (ADMM) (Boyd et al., 2011) algorithm to solve the optimization efficiently. A brief introduction of the ADMM is given in the Appendix A1.

Reformulation of ERM

We first reformulate the ERM (9) so that the ADMM algorithm can be applied smoothly. Note that the evaluation of the functional margins $\langle W_{y_i}, f(X_i) \rangle$ consists of only linear operations. We construct a big matrix $\mathbf{A}$, such that the inner product can be simplified as one matrix multiplication, i.e.,

$$\langle W_{y_i}, f(X_i) \rangle = \sum_{k=1}^{K-1} W_{y_i, k} (\langle X_i, \beta_K \rangle + \beta_{k,0}) = \mathbf{A}_{i, \cdot} \cdot \beta \text{ for } i \in [n], \quad (10)$$

where $\mathbf{A}_{i, \cdot}$ denotes the i -th row of the matrix $\mathbf{A}$. The details for constructing such a matrix $\mathbf{A}$ can be found in Appendix A2.

The penalty term in (9) consists of a sum of two L_1 norms of vectors, and thus can be simplified as

$$\|\mathbf{B}\beta\|_1 = \lambda_1 \|\beta\|_1 + \lambda_2 \|\mathbf{C}\beta\|_1,$$

where $\mathbf{B}^T = [\lambda_1 \mathbf{I}, \lambda_2 \mathbf{C}^T]$. With a little bit of adjustment to the notations, we use $\mathbf{I}$ to denote the identity matrix here. Furthermore, we reconstruct the differencing matrix $\mathbf{C}$ to a circulant matrix $\tilde{\mathbf{C}}$ by adding some additional rows, and define $\tilde{\mathbf{B}}^T = [\lambda_1 \mathbf{I}, \lambda_2 \tilde{\mathbf{C}}^T]$ accordingly. Under this reformulation, the matrix $(\mathbf{I} + \tilde{\mathbf{B}}^T \tilde{\mathbf{B}})$ becomes a block circulant with a circulant block matrix and can be efficiently inverted by using the fast Fourier transform (FFT) (Chan et al., 1993).

For masked images, we introduce a recovering matrix $\mathbf{R}$ according to the masking matrix to recover the 3-D imaging structure with all the grid points in the space. A selection matrix $\mathbf{M}$ is then introduced to rule out the augmented rows added in $\tilde{\mathbf{B}}$ and force the regions outside the mask to zeros. Therefore, we have

$$\text{FL}(\beta) = \|\mathbf{M}\tilde{\mathbf{B}}\mathbf{R}\beta\|_1.$$

The EMR is then reformulated as

$$\min_{\beta \in \mathbb{R}^{(K-1)(p+1)}} \left\{ \sum_{i=1}^n l(\mathbf{A}\beta)_i + \|\mathbf{M}\tilde{\mathbf{B}}\mathbf{R}\beta\|_1 \right\}.$$

We further introduce some auxiliary constants and artificial variables to reformulate the problem in a desired form for the ADMM. This leads to our final ERM formulation as follows:

$$\min_{\beta, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3} \sum_{i=1}^n l(\mathbf{v}_{1i}) + \|\mathbf{M}\mathbf{v}_3\|_1 \quad (11)$$

Subject to $\mathbf{v}_1 = \mathbf{A}\beta$, $\mathbf{v}_2 = \mathbf{R}\beta$, and $\mathbf{v}_3 = \tilde{\mathbf{B}}\mathbf{v}_2$.

Specifically, we set $\mathbb{X}^T = [\beta^T, \mathbf{v}_3^T]$, $\mathbb{Y}^T = [\mathbf{v}_1^T, \mathbf{v}_2^T]$, $g_1(\mathbb{X}) = \|\mathbf{M}\mathbf{v}_3\|_1$ and $g_2(\mathbb{Y}) = \sum_{i=1}^n l(\mathbf{v}_{1i})$, and denote

$$\mathbb{A}_1 = \begin{pmatrix} \mathbf{A} & 0 \\ \mathbf{R} & 0 \\ 0 & \mathbf{I} \end{pmatrix} \text{ and } \mathbb{A}_2 = \begin{pmatrix} -\mathbf{I} & 0 \\ 0 & -\mathbf{I} \\ 0 & -\tilde{\mathbf{B}} \end{pmatrix},$$

and then the updating rules for the ADMM can be adopted smoothly for our problem.

Closed-form solutions for the subproblems

We first demonstrate the solution of the optimization in the $\times$ block, which contains the following two subproblems:

$$\beta^{t+1} = \underset{\beta}{\operatorname{argmin}} \left\{ \left\| \mathbf{A}\beta - \mathbf{v}_1^t + \mathbf{u}_1^t \right\|_2^2 + \left\| \mathbf{R}\beta - \mathbf{v}_2^t + \mathbf{u}_2^t \right\|_2^2 \right\}, \quad (12)$$

$$\mathbf{v}_3^{t+1} = \underset{\mathbf{v}_3}{\operatorname{argmin}} \left\{ \left\| \mathbf{M}\mathbf{v}_3 \right\|_1 + \frac{\rho}{2} \left\| \mathbf{v}_3 - \tilde{\mathbf{B}}\mathbf{v}_2^t + \mathbf{u}_3^t \right\|_2^2 \right\}. \quad (13)$$

Solution for β :

The optimization of β in (12) is a quadratic minimization problem, which has a closed-form solution:

$$\beta^{t+1} = \mathbf{K}^t - \mathbf{A}^T (\mathbf{I} - \mathbf{A}\mathbf{A}^T)^{-1} \mathbf{A}\mathbf{K}^t = \mathbf{K}^t - \mathbf{H}_L \mathbf{H}_R^t, \quad (14)$$

where $\mathbf{K}^t = \mathbf{A}^T(\mathbf{v}_1^t - \mathbf{u}_1^t) + \mathbf{R}^T(\mathbf{v}_2^t - \mathbf{u}_2^t)$ and $\mathbf{H}_R^t = \mathbf{A}\mathbf{K}^t$. Moreover, $\mathbf{H}_L = \mathbf{A}^T(\mathbf{I} - \mathbf{A}\mathbf{A}^T)^{-1}$ is a fixed term across all iterations, so it can be precalculated.

Solution for $\mathbf{v}_3$:

Problem 13 can be solved by a proximal algorithm, the solution of which is given by

$$\mathbf{v}_3^{t+1} = \mathbf{M} \times \operatorname{Soft}_{\rho^{-1}}(\tilde{\mathbf{B}}\mathbf{v}_2^t - \mathbf{u}_3^t) - (\mathbf{I} - \mathbf{M})(\tilde{\mathbf{B}}\mathbf{v}_2^t - \mathbf{u}_3^t), \quad (15)$$

where $\operatorname{Soft}(\cdot)$ is a component-wide *soft thresholding* operator (Parikh and Boyd, 2013), denoted by $\mathbf{Soft}_{\lambda}(\mathbf{v}) = ((v_j - \lambda)_+ - (-v_j - \lambda)_+)_j$ in which $(x)_+ = \max\{x, 0\}$.

Next, we demonstrate the optimization of the $\mathbb{Y}$ block, which involves two variables $\mathbf{v}_1$ and $\mathbf{v}_2$. We apply the ADMM algorithms, and decompose it into the following two subproblems:

$$\mathbf{v}_1^{t+1} = \underset{\mathbf{v}_1}{\operatorname{argmin}} \left\{ \sum_{i=1}^n l(\mathbf{v}_{1i}) + \frac{\rho}{2} \left\| \mathbf{A}\beta^{t+1} - \mathbf{v}_1 + \mathbf{u}_1^t \right\|_2^2 \right\}, \quad (16)$$

$$\mathbf{v}_2^{t+1} = \underset{\mathbf{v}_2}{\operatorname{argmin}} \left\{ \left\| \mathbf{R}\beta^{t+1} - \mathbf{v}_2 + \mathbf{u}_2^t \right\|_2^2 + \left\| \mathbf{v}_3^{t+1} - \tilde{\mathbf{B}}\mathbf{v}_2 + \mathbf{u}_3^t \right\|_2^2 \right\}. \quad (17)$$

Solution for $\mathbf{v}_1$:

The optimization of $\mathbf{v}_1$ in (16) can be solved component-wisely by applying the Newton-Raphson method, i.e.,

$$\mathbf{v}_{1i}^{t+1} = \mathbf{v}_{1i}^t - \frac{l'(\mathbf{v}_{1i}^t) + \rho(\mathbf{v}_{1i}^t - (\mathbf{A}_i \beta^{t+1} + \mathbf{u}_{1i}^t))}{l''(\mathbf{v}_{1i}^t + \rho)}, \text{ for } i = [n], \quad (18)$$

where $l'(\cdot)$ and $l''(\cdot)$ are the first- and second-order derivatives of the loss function $l(\cdot)$, which are given as follows:

$$l'(u) = \begin{cases} -1 & \text{if } u < 0 \\ -e^{-u} & \text{if } u \geq 0 \end{cases} \text{ and } l''(u) = \begin{cases} 0 & \text{if } u < 0 \\ e^{-u} & \text{if } u \geq 0 \end{cases}.$$

To ensure convergence, we need to conduct multiple iterations in every Newton step. In our implementation, we only perform 1 iteration, which has been shown to result in sufficiently good convergence in practice.

Solution for $\mathbf{v}_2$:

The optimization of $\mathbf{v}_2$ in (17) is a standard quadratic programming problem, which has a closed-form solution:

$$\mathbf{v}_2^{t+1} = (\mathbf{I} + \tilde{\mathbf{B}}^T \tilde{\mathbf{B}})^{-1} \left\{ (\mathbf{R} \beta^{t+1} + \mathbf{u}_2) + \tilde{\mathbf{B}}^T (\mathbf{v}_3^{t+1} + \mathbf{u}_3) \right\}.$$

The direct inversion of the matrix $\mathbf{I} + \tilde{\mathbf{B}}^T \tilde{\mathbf{B}}$ may not be feasible due to the extra high dimensionality. We make use of its block circulant structure and solve the problem in $\mathbf{v}_2$ by FFT at a cost of $O(n \log n)$ operations (Afonso et al., 2010). Specifically, we have

$$\mathbf{v}_2^{t+1} = \text{ifft}(\text{fft}((\mathbf{R} \beta^{t+1} + \mathbf{u}_2) + \tilde{\mathbf{B}}^T (\mathbf{v}_3^{t+1} + \mathbf{u}_3)) \div \text{fft}(\Gamma_1)), \quad (19)$$

where fft and ifft denote the 3-D FFT and inverse FFT operators, respectively, “ $\div$ ” denotes the element-wise division, and Γ_1 is the first column of matrix $\mathbf{I} + \tilde{\mathbf{B}}^T \tilde{\mathbf{B}}$.

A complete ADMM updating procedure is summarized in Algorithm 1. We list all the involved parameters in Table 1 for a convenient reference. The primal updates are discussed above. The dual updates are directly derived from the general updating rule of the ADMM algorithm. We conduct the primal and dual updates alternatively until the prespecified convergence criteria are satisfied. In particular, we check the relative change in the estimated coefficient β , and stop the algorithm if the total number of iterations exceeds a prespecified bound or the relative change is below a certain threshold, ε , i.e.,

$$\frac{\|\beta^{t+1} - \beta^t\|}{\|\beta^t\|} \leq \varepsilon. \quad (20)$$

Algorithm 1. ADMM algorithm for SMAC-I/II

Initialize primal variables $\beta, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ as $\mathbf{0}$.

Initialize dual variables $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3$ as $\mathbf{0}$.

Set $t = 0$, assign $\lambda_1, \lambda_2 \geq 0$.

Precompute $\mathbf{H}_L = \mathbf{A}^T(\mathbf{I} - \mathbf{A}\mathbf{A}^T)^{-1}$

while $t \leq t_{max}$ **do**

Primal update:

$$\beta^{t+1} = \mathbf{K}^t - \mathbf{H}_L \mathbf{A} \mathbf{K}^t \quad (14a)$$

$$\mathbf{v}_3^{t+1} = \mathbf{M} \times \text{Soft}_{\frac{1}{\rho}} \left(\tilde{\mathbf{B}} \mathbf{v}_2^t - \mathbf{u}_3^t \right) - \left(\mathbf{I} - \mathbf{M} \right) \left(\tilde{\mathbf{B}} \mathbf{v}_2^t - \mathbf{u}_3^t \right) \quad (15a)$$

$$\mathbf{v}_{1i}^{t+1} = \mathbf{v}_{1i}^t - \frac{l'(\mathbf{v}_{1i}^t) + \rho(\mathbf{v}_{1i}^t - (\mathbf{A}_i \beta^{t+1} + \mathbf{u}_{1i}^t))}{l''(\mathbf{v}_{1i}^t) + \rho}, \text{ for } i = [n], \quad (18a)$$

$$\mathbf{v}_2^{t+1} = \text{ifft} \left(\frac{\text{fft}(\mathbf{R} \beta^{t+1} + \mathbf{u}_2) + \tilde{\mathbf{B}}^T(\mathbf{v}_3^{t+1} + \mathbf{u}_3^t)}{\text{fft}} (\Gamma_1) \right) \quad (19a)$$

Dual update:

$$\mathbf{u}_1^{t+1} = \mathbf{A} \beta^{t+1} - \mathbf{v}_1^{t+1} + \mathbf{u}_1^t$$

$$\mathbf{u}_2^{t+1} = \mathbf{R} \beta^{t+1} - \mathbf{v}_2^{t+1} + \mathbf{u}_2^t$$

$$\mathbf{u}_3^{t+1} = \mathbf{v}_3 - \tilde{\mathbf{B}} \mathbf{v}_2^{t+1} + \mathbf{u}_3^t$$

Convergence criteria:

if $\|\beta^{t+1} - \beta^t\| / \|\beta^t\| > \varepsilon$ **then**
 $t = t + 1$

else

break

return $\beta = \beta^{t+1}$

end if

end while

Simulation of synthetic data

To illustrate the finite sample performance of SMAC, we conducted simulation studies in both binary and multi-category cases.

Generation of the synthetic data

In Simulation I, we simulated 2 classes of images of size $20 \times 20 \times 10$. The true signals for each class are denoted as θ_1 and θ_2 (see Fig. 1), where θ_1 has two ROIs and θ_2 has three ROIs. The discriminating region between the two classes is the ROI represented by the region of the black triangular prism in the center, which contains 75 voxels in total. The image intensities in the three ROIs are 0, 1 and 2, respectively.

In Simulation II, we considered classifying three classes of images. The image size is $32 \times 32 \times 4$, and the true signals are θ_1 , θ_2 and θ_3 , which are graphically illustrated in Fig. 2. The image intensities are 0 in the black regions and 1 in the white regions. The discriminating regions among the three classes located in the first and second diagonal blocks are marked in the red boxes.

We generated noisy imaging samples by adding independent Gaussian noise at each voxel of the true signals, i.e., if the i -th image belongs to the k -th category, the associated noisy sample is given as

$$X_i(t) = \theta_k(t) + \varepsilon_i(t) \text{ for all } t \in \mathcal{D} \text{ and } i \in [n], \quad (21)$$

where $\varepsilon_i(t) \stackrel{iid}{\sim} N(0, \sigma^2)$ represents the Gaussian noise. For both simulation studies, we set $\sigma = 2$ for all samples.

Application: classification of MRI images from ADNI data

For the real data applications, we analyzed data from the ADNI study, a large-scale multi-site study that has collected MRI and PET images, CSF, and blood biomarkers, among other patient data. In AD, the most common form of dementia, the affected individual progressively develops disabilities in memory, language, and behavior, and the disease eventually results in death. A key goal of the ADNI study is to develop more sensitive and accurate biomarkers for the early detection of AD. The participants in the ADNI study include cognitively NCs, individuals with amnesic mild cognitive impairment (MCI), and subjects with AD. More information about this study can be found at the ADNI website (<http://adni.loni.usc.edu/>).

Participants

In this paper, we used a subset of baseline T1-weighted images from the ADNI study. After removing images with low quality, we obtained a dataset consisting of 749 samples (209 NC, 361 MCI and 179 AD). Table 2 summarizes the demographic information of all the subjects in our data analysis.

Image acquisition and processing

All images were preprocessed by a standard procedure (Guo et al., 2014), including anterior commissure and posterior commissure correction, N2 bias field correction, skull-stripping, intensity inhomogeneity correction, cerebellum removal, segmentation, and registration. We generated RAVENS-maps for the whole brain, using the deformation field obtained during

registration (Davatzikos et al., 2001) and obtained 749 images of size $128 \times 128 \times 128$. Considering that the variability of age, gender and whole-brain volume among different subjects may affect the classification results, we first removed those factors by fitting linear regression models at each voxel, and then built the classification model based on the residual images of these linear models.

Results

Comparison, tuning parameter selection and cross-validation

The proposed SMAC is designed to handle whole-brain volumetric data and detect disease-related regions without any prior spatial knowledge. To evaluate the performance of SMAC in these two tasks, we compared our method with other classifiers that can handle high-dimensional whole-brain volumetric data without any pre-screening procedure, and which also have the ability to yield volumetric coefficient images in the same space of the covariates. Under this guidance for comparison, we chose the following classifiers for neuroimaging classification: logistic regression using elastic-net regularization (EN-LR) (Casanova et al., 2011), logistic regression with the GraphNet penalty (GN-LR) (Grosenick et al., 2013) and SSVM with an FL penalty (Watanabe et al., 2014). Since SSVM was originally designed only for binary problems, we did not include it in the multi-category problems. To distinguish between SMAC with TV-I and TV-II penalties, we respectively denote them as SMAC-I and SMAC-II.

All the methods mentioned above involve two tuning parameters, λ_1 and λ_2 . For consistent comparison, we denoted λ_1 as the tuning parameter of the sparse penalty terms for all methods. In SSVM, SMAC-I and SMAC-II, we denoted λ_2 as the tuning parameter of the total variation terms, whereas in EN-LR and GN-LR, we defined λ_2 as the parameter of the L_2 norm penalty. We conducted a grid search to select the best pair of the two parameters across a 21 by 21 log-based grid for the synthetic data, i.e., $\lambda_1 \otimes \lambda_2 \in \{0, 2^{-14}, 2^{-13}, \dots, 2^5\}^{\otimes 2}$ and a smaller grid of $\lambda_1 \otimes \lambda_2 \in \{0, 2^{-13}, 2^{-11}, \dots, 2^3, 2^5\}^{\otimes 2}$ for the real data.

For the analysis of the synthetic data, a data-rich scenario, we independently generated 30 training, 30 validation and 300 test samples for each class according to (21), which yielded 60 training, 60 validation and 600 test samples in Simulation I and 90 training, 90 validation and 900 test samples in Simulation II. We used the training samples to build models for each combination of λ_1 and λ_2 , and evaluated the models on the validation samples to calculate the tuning classification accuracy and area under the curve (AUC) in the associated receiver operating characteristic (ROC) analysis. Based on the validation results, we picked the models with the highest classification accuracy. If ties occurred, we chose the models with highest AUC among them. If we still obtained multiple models, the one with a larger spatial penalty (λ_2) was selected as our final model. We applied the final model to the test samples to evaluate the classification performance. To validate the stability of the methods, we repeated the experiments for 50 iterations, and reported the means and standard deviations of the results.

For the real data analysis, we applied a stratified sampling on the whole dataset and split it into training (60%), validation (20%) and test (20%) sets, so that the proportions of NC,

MCI and AD subjects were similar across the different sets. We used a validation and evaluation procedure that was similar to what we used in the simulation study. We repeated the above random split 30 times and recorded the means and standard deviations of the results.

Results from synthetic data analysis

Cross-validation and tuning results—The mean validation accuracy matrices from 50 iterations of the simulation studies are given in Fig. 3. In Simulation I (binary case), EN-LR yielded lower validation accuracy for most of the sparse estimation, i.e., $\lambda_1 \in \{2^1, \dots, 2^5\}$. SSVM yielded higher tuning accuracies for the sparse and patched estimation, i.e., $\lambda_1 \otimes \lambda_2 \in \{0, 2^{-14}, \dots, 2^{-8}\} \otimes \{2^{-5}, \dots, 2^{-3}\}$. GN-LR achieved very good validation accuracy when the sparsity and smoothness levels were relatively high, but yielded low accuracy when the sparsity level was too high, i.e., $\lambda_1 \in \{2^4, 2^5\}$. SMAC-I and SMAC-II achieved overall higher validation accuracy and were more sensitive to the change in tuning parameters. In particular, the SMAC methods were more sensitive to the penalty level of the total variation than the sparse term. This is mainly explained by the spatial smoothness assumption of the imaging data.

The results of Simulation II are similar to those of Simulation I. The sparse method EN-LR yielded low validation accuracy for most combinations of the tuning parameters. GN-LR and SMAC achieved high accuracy under a relatively high sparsity level and a moderate smoothness penalty level, i.e., $\lambda_1 \otimes \lambda_2 \in \{2^{-5}, 2^{-4}, 2^{-3}\}^{\otimes 2}$.

Receiver operating characteristic (ROC) analysis and classification accuracy

—The ROC analysis can simultaneously evaluate the true positive rate and the false positive rate for a binary classifier under different thresholds. The AUC numerically measures the performance of a classifier in the ROC analysis. When dealing with the multi-category cases, the ROC analysis can be implemented using the “one vs. the rest” strategy, i.e., transforming it into multiple binary problems. We conducted the ROC analysis for both binary and multi-category problems, randomly picked one result from the 50 iterations, and plotted the associated ROC curves; see Figs. 4 and 5. The numerical results for all iterations are summarized in Tables 3 and 4.

In the binary classification example, SMAC-I achieved the highest classification accuracy of 96.52% and the largest AUC of 99.58%, followed by an accuracy of 96.17% and an AUC of 99.39% from SMAC-II. GN-LR and SSVM yielded accuracies of 93.04% and 92.23%, and AUC values of 98.11% and 97.91%, respectively. EN-LR achieved an accuracy of 73.75% and an AUC of 81.70%.

In the multi-category cases, SMAC-I and SMAC-II yielded the highest respective accuracies of 94.70% and 94.19%, as well as the largest AUC values in all three classes. GN-LR achieved an accuracy of 92.17%. EN-LR yielded an accuracy of 64.89% and had the smallest AUC values in all three classes.

In both simulation studies, the spatial methods were more stable in terms of the classification results and yielded smaller standard deviations among the 50 iterations. The

sparse method EN-LR delivered sparse estimation consisting of isolated voxels, and thus yielded unstable models. In particular, its variable selection results varied a lot in different iterations.

Visualization and interpretation of coefficient images—We plotted all the coefficient images to illustrate the estimation and identification of those critical regions for classifying the samples. The plot of the coefficient images in Simulation I (see Fig. 6) reveals that EN-LR yielded a sparse coefficient image, consisting of isolated voxels; whereas all the other spatial penalized methods produced smooth coefficient images, clearly indicating the triangular discriminating region in the center. SSVM and SMAC-I both yielded clear boundaries between the predictive and irrelevant regions. SSVM contained many false positive voxels in the background; whereas SMAC-I had a “clean” background. GN-LR yielded smooth coefficient images with blurred boundaries around the triangular region and also contained some false positives in the background. SMAC-II yielded a similarly smooth coefficient image with many fewer false positives.

In the multi-category example, we illustrated the reconstructed coefficient images (defined in Equation (6)) from SMAC-I/II and compared them with the penalized logistic regression methods (EN-LR and GN-LR). Since the three coefficients for each method summed to zero, we only displayed first two of them, i.e. $\hat{\beta}_1$ and $\hat{\beta}_2$ (see Fig. 7). The estimated coefficient images obtained from EN-LR consist of isolated voxels. GN-LR and SMAC-II yielded smooth patched estimations, but with a blurred boundary. The coefficient images from SMAC-I clearly captured the first and second diagonal block regions of the checkerboard image, which are the most critical regions for discriminating the three classes.

Accurately capturing the key discriminating regions is a requirement of a good image classifier. In both simulation studies, the sparsity-only classifier EN-LR underperformed due to the ignorance of the spatial structure. GN-LR and SMAC-II tended to yield smooth critical regions in which the imaging intensities continuously varied across voxels. SSVM and SMAC-I were able to capture the critical regions with clear boundaries. SMAC-I and SMAC-II achieved fewer false positives in the irrelevant regions, while the other methods contained either isolated or patchy false positives in the background. For these particular synthetic data, SMAC-I delivered the most competitive performance. This was mainly due to the assumption of patchy constant patterns in the discriminating regions. SMAC-II may have potential advantages when those regions have continuously varying intensities.

Model sensitivity on training sample size and noise level—To further analyze the stability of the proposed methods, we conducted a comprehensive sensitivity analysis on the sample size and noise level. In particular, the sample size analysis was done by repeating the experiment in Simulation I with the training sample size ranging from 10 to 100. The validation and test sample sizes were not changed, and the noise level remained the same for different sample sizes, i.e., $\varepsilon_i(t) \stackrel{iid}{\sim} N(0, 4)$. A similar model selection procedure as that used in Simulation I was used, and we report the test results in Table 5 and Fig. 8. The noise level sensitivity analysis was done by fixing the training sample size ($n = 30$) and varying the

standard deviation of the noise added to each voxel from $\sigma = 1$ to 4. The test results are summarized in Table 6 and Fig. 9.

From the sensitivity analysis, we conclude that the proposed SMAC methods can achieve high accuracies and AUCs with very limited training samples, e.g., $n = 50$, and yield very competitive performance in the cases of mid-noise levels, e.g., $\sigma \in (2, 3)$.

Results from ADNI data—We conducted both binary and multi-category classification experiments using the ADNI data. In particular, we classified all possible pairs of the three classes as binary problems (NC vs AD, NC vs MCI and MCI vs AD) and identified AD, MCI and NC simultaneously as a three-category problem. The classification accuracies are presented in Tables 7 and 8. After we obtained the best tuning parameters from the 30 iterations of the three-way split, we refitted the model using all the data with the selected parameters and registered the coefficient images to the Montreal Neurological Institute (MNI)-152 template (Fonov et al., 2011). A plot of these coefficients in the orthogonal views is provided in Figs. 10 and 11.

ROC analysis and classification accuracy—In the classification problem of NC vs AD, SMAC-I and SMAC-II achieved the highest two accuracies of 89.12% and 88.33% respectively. The other three methods yielded similar accuracies between 86% and 87%. In the classification of MCI vs AD and NC vs MCI, the overall accuracies were lower. This may be partially explained by the uncertainty involved in the cognitive test for identifying MCI and the heterogeneity within the MCI group. EN-LR and SMAC-I/II yielded accuracy values that were very close in these tasks. SSVM was outperformed by the other methods, and could not capture informative signals in the classification of NC vs MCI. Notice that GN-LR achieved the highest values of AUC in all three binary classification problems. This is explained by the merit of the logistic loss in terms of estimating the “soft” class label, i.e. the associated probability (Liu et al., 2011). SMAC-I/II also yielded very competitive AUC values (second best in all three problems).

For the simultaneous classification of NC, MCI and AD, the classification accuracies are lower than those for the binary cases. SMAC-I/II yielded higher accuracies (53.22% and 52.68%) compared to those achieved by EN-LR and GN-LR (49.32% and 49.75%). The AUC values for MCI (AUC2) were lower than the ones for NC and AD, which was consistent with the results in the binary cases. SMAC-II achieved the best and second best values for AUC1 and AUC3, indicating a better detection rate for NC and AD.

Clinically meaningful coefficient images—Different from our synthetic imaging data, the MRI images of human brains are much more complex. Due to heterogeneity across subjects and the potential bias in the registration process, the boundaries between the discriminating regions and the background may not be as sharp as they are in the synthetic data. The patchy patterns may not be a perfect assumption for this case, but still help the classifiers to recover the predictive regional signals. From the plots in Figs. 10 and 11, we can clearly see consistent patterns across the coefficient images from different spatial methods. The sparse method EN-LR delivers ultra-sparse estimation that is difficult to interpret biologically. Notice that, among all the spatial methods, SMAC-II can recover more

smooth and patchy signals, while screening out the irrelevant regions in the brain. This will make it easier to identify ROIs in the coefficient images of SMAC-II.

Comparing Figs. 10 and 11, we can see that for each spatial method, the effective regions of β in Fig. 10 and the first coefficient β_1 in Fig. 11 are relatively consistent, but the intensity values have the opposite signs, i.e., the regional effects are opposite. This is mainly because the positive class label in the binary problem is AD while in the multi-category problem, class label 1 is associated with NC.

By overlaying the coefficient images from SMAC-I/II on the MNI-152 ROI template, we are able to identify several significant discriminating regions, such as the frontal gyrus, hippocampus, and right fornix. Many papers in the existing literature have shown that these regions are potentially related to the development of MCI and AD. For instance, the hippocampal region is involved in memory processes that deteriorate with the development of AD. The structure of the hippocampus is altered by the degenerative processes associated with AD, and loss of the hippocampal volume occurs at a rate that is approximately two to four times faster in patients with AD than in age-matched healthy controls (West et al., 1994; Dubois et al., 2014).

Computational considerations—In the MATLAB implementation of our algorithms, most of the computation is realized through matrix operations. For moderate image sizes (e.g., total number of voxels less than 10^4), our methods converge very fast compared to the others. For ultra-high-dimensional imaging data (e.g., total number of voxels greater than 10^6), the matrix operations require more memory usage. Furthermore, for all the classifiers used in our comparison, the regularization parameters highly affect the convergence and computational cost of the algorithms. We ran all the programs on the same type of computer (Intel Xeon E5-2643 v3 @ 3.40 GHz) with the same random-access memory (8 GB DDR3 at 1600 MHz). All algorithms were set with the same maximum number of iterations ($t_{max} = 1500$) and convergence threshold ($\epsilon = 5 \times 10^{-5}$) as defined in (20). We plotted the mean computational time from all 5 classifiers among 50 iterations in Simulation I (see Fig. 12). EN-LR required the shortest time for this classification problem. The variation in the computational time was very small. This was mainly due to the simplicity of the EN-LR model. SMAC-I required the second shortest computational time, followed by SMAC-II. This was because the second-order total variation involved computation of the discrete Hessian operators, which had larger sizes than the gradient operators in SMAC-I. GN-LR also yielded very competitive computational speed. SSVM was out-performed by the other classifiers in terms of the computational speed. This was mainly due to the splitting scheme in its ADMM algorithm and the heavy computational load in optimization involving the non-smoothing hinge loss.

Discussion

In this paper, we propose a SMAC for neuroimaging classification. Our method achieves the desired spatial sparsity and smoothness in the coefficient images via imposing the FL penalty. It improves the accuracy in both binary and multi-category classification problems.

Both the simulation studies and the real data application demonstrate the usefulness of the proposed method.

Numerous classification studies in the literature have used the ADNI data, but their data collection and evaluation procedures may vary significantly. A direct comparison of the results may not be a reasonable way to evaluate the methods. For example, Dukart et al. (2011) achieved 100% accuracy on the classification of NC vs AD, while we obtained 89.12% accuracy for the same problem. However, their study assessed only 13 NC and 21 AD subjects; whereas our study assessed the 749 participants in the ADNI study. Moreover, they used pre-computed ROI statistics from both MRI and fluorodeoxyglucose-PET images as predictors; whereas we directly classified the baseline MRI data and automatically extracted the regional information during the estimation procedure. An advantage of our proposed method is that we can handle imaging data with limited pre-processing, and still produce reasonably good results. This can be valuable when the prior knowledge of spatial segmentation is not available.

We introduce an efficient algorithm using ADMM to solve the corresponding large-scale optimization problem in our method. Specifically, we propose a novel splitting scheme in ADMM and reduce the complexity of optimization. As a result, our algorithm performs more efficiently than the ADMM algorithms in the existing literature, such as Ye and Xie (2011) and Watanabe et al. (2014). Moreover, the proposed algorithm is very flexible and can be applied to solve various other prediction problems within the *loss + penalty* framework. We have included the implementation of the squared error loss in our package, which allows users to perform spatial regularized high-dimensional regression analysis. Details about this extension are included in Appendix A3.

One potential limitation of the proposed method is the underlying assumption of the spatially clustered patterns in the true coefficient images. This is a reasonable assumption in most neuroimaging applications. However, if the overall predictive effect is scattered around most of the imaging space, this method can be inefficient due to the complexity of the optimization problem.

There are several possible interesting extensions of the proposed method for future exploration. For example, all linear classifiers are built based on the assumption that the images are perfectly aligned and the predictive regions are consistent across all the subjects within the same class. These assumptions can be violated in practice, both due to the non-negligible registration error and the heterogeneous structures within the population. The estimation and predictive performance can be highly affected by this issue. One possible future research direction is to handle this heterogeneity problem.

Acknowledgment

Data collection and sharing for this project was funded by the Alzheimer's Disease Neuroimaging Initiative (ADNI) (National Institutes of Health Grant U01 AG024904) and DOD ADNI (Department of Defense award number W81XWH-12-2-0012). ADNI is funded by the National Institute on Aging, the National Institute of Biomedical Imaging and Bioengineering, and through generous contributions from the following: Alzheimers Association; Alzheimers Drug Discovery Foundation; Araclon Biotech; BioClinica, Inc.; Biogen Idec Inc.; Bristol-Myers Squibb Company; Eisai Inc.; Elan Pharmaceuticals, Inc.; Eli Lilly and Company; EuroImmun; F. Hoffmann-La Roche Ltd and its affiliated company Genentech, Inc.; Fujirebio; GE Healthcare; IXICO Ltd.; Janssen Alzheimer

Immunotherapy Research & Development, LLC.; Johnson & Johnson Pharmaceutical Research & Development LLC.; Medpace, Inc.; Merck & Co., Inc.; Meso Scale Diagnostics, LLC.; NeuroRx Research; Neurotrack Technologies; Novartis Pharmaceuticals Corporation; Pfizer Inc.; Piramal Imaging; Servier; Synarc Inc.; and Takeda Pharmaceutical Company. The Canadian Institutes of Health Research are providing funds to support ADNI clinical sites in Canada. Private sector contributions are facilitated by the Foundation for the National Institutes of Health (www.fnih.org). The grantee organization is the Northern California Institute for Research and Education, and the study is coordinated by the Alzheimer's Disease Cooperative Study at the University of California, San Diego. ADNI data are disseminated by the Laboratory for Neuro Imaging at the University of Southern California.

★ This work was partially supported by NIH grants MH086633, MH092335, GM126550 and CA142538, NSF grants SES-1357666, DMS-1407655, DMS-1407241 and IIS-1632951, the grant RR150054 from the Cancer Prevention Research Institute of Texas, and the endowed Bao-Shan Jing Professorship in Diagnostic Imaging. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NIH. The readers are welcome to request reprints from Dr. Hongtu Zhu. hzh5@mdanderson.org; Phone: 346-8140191. "Data used in preparation of this article were obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (adni.loni.usc.edu). As such, the investigators within the ADNI contributed to the design and implementation of ADNI and/or provided data but did not participate in the data analysis or writing of this report. A complete listing of ADNI investigators can be found at http://adni.loni.usc.edu/wp-content/uploads/how-to-apply/ADNI_Acknowledgement_List.pdf"

Appendix A1. Alternative Direction Method of Multipliers

The ADMM algorithm (Boyd et al., 2011; Mota et al., 2011) was developed to handle large-scale convex optimization problems with the following separable and constrained structure:

$$\min_{\mathbb{X}, \mathbb{Y}} g_1(\mathbb{X}) + g_2(\mathbb{Y}) \text{ subject to } \mathbb{A}_1 \mathbb{X} + \mathbb{A}_2 \mathbb{Y} = 0, \quad (22)$$

where $\mathbb{X} \in \mathbb{R}^p$ and $\mathbb{Y} \in \mathbb{R}^q$ are unknown parameters, $g_1(\mathbb{X})$ and $g_2(\mathbb{Y})$ are two closed convex functions, and $\mathbb{A}_1 \in \mathbb{R}^{m \times p}$ and $\mathbb{A}_2 \in \mathbb{R}^{m \times q}$ represent m linear constraints on $\mathbb{X}$ and $\mathbb{Y}$, respectively. ADMM solves (22) by breaking them into smaller and simpler subproblems and solving them alternatively. Specifically, for the $t + 1$ iteration,

$$\begin{aligned} \mathbb{X}^{t+1} &= \arg \min_{\mathbb{X}} \left\{ g_1(\mathbb{X}) + \frac{\rho}{2} \left\| \mathbb{A}_1 \mathbb{X} + \mathbb{A}_2 \mathbb{Y}^t + \mathbf{u}^t \right\|_2^2 \right\}, \\ \mathbb{Y}^{t+1} &= \arg \min_{\mathbb{Y}} \left\{ g_2(\mathbb{Y}) + \frac{\rho}{2} \left\| \mathbb{A}_1 \mathbb{X}^{t+1} + \mathbb{A}_2 \mathbb{Y} + \mathbf{u}^t \right\|_2^2 \right\}, \\ \mathbf{u}^{t+1} &= \mathbb{A}_1 \mathbb{X}^{t+1} + \mathbb{A}_2 \mathbb{Y}^{t+1} + \mathbf{u}^t, \end{aligned}$$

where ρ is the augmented Lagrangian parameter, $\mathbf{u}$ is a vector of dual variables, and $\|\cdot\|_2$ denotes the L_2 Euclidean norm. The choice of ρ affects the convergence rate of the algorithm (Boyd et al., 2011), and remains an open question in the literature. We implement our algorithm with $\rho = 1$, but it can be tuned in practice.

A2. The construction of matrix A

To simplify the inner product in the loss function, we need to construct a big matrix A to summarize all the linear operations. First, we denote

$$W_{y_i} = \left(W_{y_i,1}, W_{y_i,2}, \dots, W_{y_i,K-1} \right)^T \in \mathbb{R}^{K-1}$$

and construct a matrix $\tilde{W}_Y$ consisting of a stack of diagonal matrices, i.e.,

$$\tilde{W}_Y = \left[W_{Y,1}, W_{Y,2}, \dots, W_{Y,K-1} \right].$$

where $W_{Y,l} = \text{diag}\{W_{y_1,l}, \dots, W_{y_n,l}\}$ for $l = 1, \dots, K-1$. Moreover, we denote $\tilde{X} = \text{diag}\{X, \dots, X\}$ as a matrix consisting of $K-1$ copies of the original covariate matrix on the diagonal. In particular, the columns of 1's are added at the first column of the covariate matrix to include the intercepts in the computation. Then, we define $A = \tilde{W}_Y \tilde{X}$, and it can be verified that

$$A\beta = \begin{bmatrix} \left\langle W_{y_1}, f(X_1) \right\rangle \\ \vdots \\ \left\langle W_{y_n}, f(X_n) \right\rangle \end{bmatrix}.$$

A3. Spatial regularized regression

The ADMM algorithm proposed in this paper is quite flexible and can be extended to solve other problems. In this section, we introduce the extension of our algorithm to solve a spatial regularized regression problem.

For the regression problem, the response variable y_i can be a continuous measure of a certain clinical index. Denote the covariate image as X_i . The regularized regression problem is given by

$$\hat{\beta} = \arg \min \left\{ \frac{1}{2} \sum_{i=1}^n (y_i - X_i \beta)^2 + FL(\beta) \right\}.$$

This is analogous to equation (9), by letting $K=2$ and applying the square loss, i.e., $\ell(u) = u^2$. Both the first- and second-order total variations can be applied here.

We adopt the reformulation of equation (9) and construct a similarly constrained optimization to problem (11), i.e.,

$$\min_{\beta, v_1, v_2, v_3} \frac{1}{2} \sum_{i=1}^n (y_i - v_{1i})^2 + \left\| M v_3 \right\|_1$$

Subject to $v_1 = X\beta$, $v_2 = R\beta$, and $v_3 = \tilde{B}v_2$.

Here, all the variable are the same as those in the algorithm for SMAC, but with fixed $K = 2$. The only change is that the loss function part becomes the squared error loss and $\mathbf{A}$ in problem (11) becomes $\mathbf{X}$. Thus, the solutions for β , $\mathbf{v}_2$ and $\mathbf{v}_3$ remain the same by setting $A = \mathbf{X}$ in (14).

The subproblem involving $\mathbf{v}_1$ becomes the following:

$$\mathbf{v}_1^{t+1} = \underset{\mathbf{v}_1}{\operatorname{argmin}} \left\{ \frac{1}{2} \|\mathbf{Y} - \mathbf{v}_1\|_2^2 + \frac{\rho}{2} \|\mathbf{A}\beta^{t+1} - \mathbf{v}_1 + \mathbf{u}_1^t\|_2^2 \right\},$$

where $\mathbf{Y} = [y_1, \dots, y_n]^T$. This is a standard quadratic programming problem for which the solution is given by

$$\mathbf{v}_1^{t+1} = \frac{1}{1 + \rho} (\mathbf{Y} + \mathbf{A}\beta^{t+1} - \mathbf{u}_1^t). \quad (23)$$

Therefore, the whole algorithm can be summarized as Algorithm 2.

Algorithm 2. ADMM algorithm for spatial regularized regression.

Initialize primal variables $\beta, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ as $\mathbf{0}$.

Initialize dual variables $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3$ as $\mathbf{0}$.

Set $t = 0$, assign $\lambda_1, \lambda_2 \geq 0$.

Set $\mathbf{A} = \mathbf{X}$ and $K = 2$.

Precompute $\mathbf{H}_L = \mathbf{A}^T(I - \mathbf{A}\mathbf{A}^T)^{-1}$

while $t \leq t_{\max}$ **do**

Primal update:

$$\begin{aligned}\beta^{t+1} &= \mathbf{K}^t - \mathbf{H}_L \mathbf{A} \mathbf{K}^t \\ \mathbf{v}_3^{t+1} &= \mathbf{M} \times \text{Soft}_{\frac{1}{\rho}} \left(\tilde{\mathbf{B}} \mathbf{v}_2^t - \mathbf{u}_3^t \right) - \left(\mathbf{I} - \mathbf{M} \right) \left(\tilde{\mathbf{B}} \mathbf{v}_2^t - \mathbf{u}_3^t \right) \\ \mathbf{v}_1^{t+1} &= \frac{1}{1+\rho} \left(\mathbf{Y} + \mathbf{A} \beta^{t+1} - \mathbf{u}_1^t \right) \\ \mathbf{v}_2^{t+1} &= \text{fft} \left(\frac{\text{fft} \left(\left(\mathbf{R} \beta^{t+1} + \mathbf{u}_2^t \right) + \tilde{\mathbf{B}}^T \left(\mathbf{v}_3^{t+1} + \mathbf{u}_3^t \right) \right)}{\text{fft}} \right) (\Gamma_1)\end{aligned}$$

Dual update:

$$\begin{aligned}\mathbf{u}_1^{t+1} &= \mathbf{A} \beta^{t+1} - \mathbf{v}_1^{t+1} + \mathbf{u}_1^t \\ \mathbf{u}_2^{t+1} &= \mathbf{R} \beta^{t+1} - \mathbf{v}_2^{t+1} + \mathbf{u}_2^t \\ \mathbf{u}_3^{t+1} &= \mathbf{v}_3 - \tilde{\mathbf{B}} \mathbf{v}_2^{t+1} + \mathbf{u}_3^t\end{aligned}$$

Convergence criteria:

if $\|\beta^{t+1} - \beta^t\| / \|\beta^t\| > \varepsilon$ **then**
 $t = t + 1$

else

break

return $\beta = \beta^{t+1}$

end if

end while

References

- Afonso MV, Bioucas-Dias JM, Figueiredo MA, 2010 Fast image recovery using variable splitting and constrained optimization. Image Process. IEEE Trans 19 (9), 2345–2356.
- Beck A, Teboulle M, 2009 A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. imaging Sci 2 (1), 183–202.
- Boser BE, Guyon IM, Vapnik VN, 1992 A training algorithm for optimal margin classifiers. In: Proceedings of the Fifth Annual Workshop on Computational Learning Theory ACM, pp. 144–152.
- Boyd S, Parikh N, Chu E, Peleato B, Eckstein J, 2011 Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends[®] Mach. Learn. 3 (1), 1–122.
- Busatto GF, Garrido GE, Almeida OP, Castro CC, Camargo CH, Cid CG, Buchpiguel CA, Furuie S, Bottino CM, 2003 A voxel-based morphometry study of temporal lobe gray matter reductions in alzheimers disease. Neurobiol. aging 24 (2), 221–231. [PubMed: 12498956]
- Casanova R, Whitlow CT, Wagner B, Williamson J, Shumaker SA, Maldjian JA, Espeland MA, 2011 High dimensional classification of structural mri alzheimers disease data based on large scale regularization. Front. neuroinformatics 5.

- Chan RH, Nagy JG, Plemmons RJ, 1993 Fft-based preconditioners for toeplitz-block least squares problems. *SIAM J. Numer. analysis* 30 (6), 1740–1768.
- Chen E, Chung P-C, Chen C-L, Tsai H-M, Chang C-I, et al., 1998 An automatic diagnostic system for ct liver image classification. *Biomed. Eng. IEEE Trans* 45 (6), 783–794.
- Colliot O, Chételat G, Chupin M, Desgranges B, Magnin B, Benali H, Dubois B, Garnero L, Eustache F, Lehericy S, 2008 Discrimination between alzheimer disease, mild cognitive impairment, and normal aging by using automated segmentation of the hippocampus 1. *Radiology* 248 (1), 194–201. [PubMed: 18458242]
- Davatzikos C, Genc A, Xu D, Resnick SM, 2001 Voxel-based morphometry using the ravens maps: methods and validation using simulated longitudinal atrophy. *NeuroImage* 14 (6), 1361–1369. [PubMed: 11707092]
- Dubois B, Feldman HH, Jacova C, Hampel H, Molinuevo JL, Blennow K, DeKosky ST, Gauthier S, Selkoe D, Bateman R, et al., 2014 Advancing research diagnostic criteria for alzheimer's disease: the iw-2 criteria. *Lancet Neurology* 13 (6), 614–629. [PubMed: 24849862]
- Dukart J, Mueller K, Horstmann A, Barthel H, Möller HE, Villringer A, Sabri O, Schroeter ML, 2011 Combined evaluation of fdg-pet and mri improves detection and differentiation of dementia. *PLoS One* 6 (3) e18111. [PubMed: 21448435]
- Fisher RA, 1936 The use of multiple measurements in taxonomic problems. *Ann. Eugen* 7 (2), 179–188.
- Fonov V, Evans AC, Botteron K, Almli CR, McKinstry RC, Collins DL, Group BDC, et al., 2011 Unbiased average age-appropriate atlases for pediatric studies. *Neuroimage* 54 (1), 313–327. [PubMed: 20656036]
- Freund Y, Schapire RE, 1997 A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci* 55 (1), 119–139.
- Friedman J, Hastie T, Tibshirani R, et al., 2000 Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *Ann. statistics* 28 (2), 337–407.
- Grosenick L, Greer S, Knutson B, 2008 Interpretable classifiers for fmri improve prediction of purchases. *Neural Syst. Rehabilitation Eng. IEEE Trans.* 16 (6), 539–548.
- Grosenick L, Klingenberg B, Greer S, Taylor J, Knutson B, 2009 Whole-brain sparse penalized discriminant analysis for predicting choice. *NeuroImage* 47, S58.
- Grosenick L, Klingenberg B, Katovich K, Knutson B, Taylor JE, 2013 Interpretable whole-brain prediction analysis with graphnet. *NeuroImage* 72, 304–321. [PubMed: 23298747]
- Guo R, Ahn M, Zhu H, 2014 Spatially weighted principal component analysis for imaging classification. *J. Comput. Graph. Statistics* (just-accepted), 00–00.
- Hastie T, Tibshirani R, Friedman J, Franklin J, 2005 The elements of statistical learning: data mining, inference and prediction. *Math. Intell* 27 (2), 83–85.
- Hinrichs C, Singh V, Xu G, Johnson SC, Initiative ADN, et al., 2011 Predictive markers for ad in a multi-modality framework: an analysis of mci progression in the adni population. *Neuroimage* 55 (2), 574–589. [PubMed: 21146621]
- Liu M, Zhang D, Shen D, Initiative ADN, et al., 2012 Ensemble sparse classification of alzheimer's disease. *NeuroImage* 60 (2), 1106–1116. [PubMed: 22270352]
- Liu Y, Yuan M, 2011 Reinforced multicategory support vector machines. *J. Comput. Graph. Statistics* 20 (4), 901–919.
- Liu Y, Zhang HH, Wu Y, 2011 Hard or soft classification? large-margin unified machines. *J. Am. Stat. Assoc* 106 (493), 166–177. [PubMed: 22162896]
- Lopez M, Ramirez J, Gorris J, Salas-Gonzalez D, Alvarez I, Segovia F, Chaves R, 2009 Neurological image classification for the alzheimer's disease diagnosis using kernel pca and support vector machines. In: *Nuclear Science Symposium Conference Record (NSS/MIC)*, 2009 IEEE IEEE, pp. 2486–2489.
- Mota JF, Xavier JM, Aguiar PM, Puschel M, 2011 A Proof of Convergence for the Alternating Direction Method of Multipliers Applied to Polyhedral-constrained Functions *arXiv preprint arXiv:1112.2295*.
- Parikh N, Boyd S, 2013 Proximal algorithms. *Found. Trends Optim* 1 (3), 123–231.

- Ramírez J, Chaves R, Górriz JM, Álvarez I, López M, Salas-Gonzalez D, Segovia F, 2009 Functional brain image classification techniques for early alzheimer disease diagnosis. In: Bioinspired Applications in Artificial and Natural Computation. Springer, pp. 150–157.
- Rudin LI, Osher S, Fatemi E, 1992 Nonlinear total variation based noise removal algorithms. *Phys. D. Nonlinear Phenom* 60 (1), 259–268.
- Rusinek H, Endo Y, De Santi S, Frid D, Tsui W-H, Segal S, Convit A, de Leon M, 2004 Atrophy rate in medial temporal lobe during progression of alzheimer disease. *Neurology* 63 (12), 2354–2359. [PubMed: 15623699]
- Tibshirani R, 1996 Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B Methodol* 267–288.
- Tibshirani RJ, 2011 The Solution Path of the Generalized Lasso. Stanford University.
- Tibshirani R, Saunders M, Rosset S, Zhu J, Knight K, 2005 Sparsity and smoothness via the fused lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol* 67 (1), 91–108.
- Van Gerven MA, Heskes T, 2012 A linear Gaussian framework for decoding of perceived images. In: 2012 Second International Workshop on Pattern Recognition in NeuroImaging IEEE, pp. 1–4.
- Watanabe T, Kessler D, Scott C, Angstadt M, Sripada C, 2014 Disease prediction based on functional connectomes using a scalable and spatially-informed support vector machine. *Neuroimage* 96, 183–202. [PubMed: 24704268]
- West MJ, Coleman PD, Flood DG, Troncoso JC, 1994 Differences in the pattern of hippocampal neuronal loss in normal ageing and alzheimer's disease. *Lancet* 344 (8925), 769–772. [PubMed: 7916070]
- Xu Y, Jack C, O'Brien P, Kokmen E, Smith G, Ivnik R, Boeve B, Tangalos R, Petersen R, 2000 Usefulness of mri measures of entorhinal cortex versus hippocampus in ad. *Neurology* 54 (9), 1760–1767. [PubMed: 10802781]
- Yamashita O, Sato M. a., Yoshioka T, Tong F, Kamitani Y, 2008 Sparse estimation automatically selects voxels relevant for the decoding of fmri activity patterns. *NeuroImage* 42 (4), 1414–1429. [PubMed: 18598768]
- Ye G-B, Xie X, 2011 Split bregman method for large scale fused lasso. *Comput. Statistics Data Analysis* 55 (4), 1552–1569.
- Yu G, Liu Y, Thung K-H, Shen D, 2014 Multi-task Linear Programming Discriminant Analysis for the Identification of Progressive Mci Individuals.
- Zhang C, Liu Y, 2014 Multicategory angle-based large-margin classification. *Biometrika* 101 (3), 625–640. [PubMed: 26538663]
- Zhu J, Hastie T, 2005 Kernel logistic regression and the import vector machine. *J. Comput. Graph. Statistics* 14 (1).
- Zhu J, Zou H, Rosset S, Hastie T, 2009 Multi-class adaboost. *Statistics its Interface* 2 (3), 349–360.
- Zhu X, Suk H-I, Shen D, 2014 Sparse Discriminative Feature Selection for Multi-class Alzheimers Disease Classification, pp. 157–164.
- Zou H, Hastie T, 2005 Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Ser. B Stat. Methodol* 67 (2), 301–320.

Fig. 1.

True signals for two classes of images in Simulation I. The left panel is the true image of class 1: the transparent and yellow regions represent the voxel values of 0 and 1, respectively. The right panel is the true image of class 2: the transparent, yellow and black regions represent 0, 1 and 2, respectively.

Fig. 2.

True signals for three classes of images in Simulation II. The three images are the top layer ($z = 1$) of the mean images for class 1, 2, and 3, respectively. White represents the voxel value of 1, and black represents 0. The four layers ($z = 1, \dots, 4$) of the true image are identical within each class. The discriminating regions are marked in red boxes.

Fig. 3.

Validation accuracies for synthetic studies. The top row of 5 panels (from left to right) respectively correspond to the validation accuracy matrices of EN-LR, GN-LR, SSVM, SMAC-I and SMAC-II for the binary synthetic data. The bottom row of 4 panels (from left to right) respectively correspond to the validation accuracy matrices of EN-LR, GN-LR, SMAC-I and SMAC-II for the multi-category synthetic data. Each entry of the matrix is the tuning accuracy for the corresponding combination of λ_1 and λ_2 . The vertical direction of the matrix represents the value of λ_1 , from top to bottom being $\{0, 2^{-14}, 2^{-13}, \dots, 2^5\}$, and the horizontal direction represents λ_2 , from left to right being $\{0, 2^{-14}, 2^{-13}, \dots, 2^5\}$.

Fig. 4. Receiver operating characteristic (ROC) analysis for the binary synthetic data based on 600 test samples.

Fig. 5.

Receiver operating characteristic (ROC) analysis for the multi-category synthetic data based on 900 test samples. Each panel represents the ROC curves evaluated using the “one-versus-the-rest” strategy.

Fig. 6. Estimated coefficient images obtained from five classification methods in Simulation I. The 5 panels display the coefficient images of EN-LR, SSVM, GN-LR, SMAC-I and SMAC-II. Each coefficient image is displayed in three respective directions: transverse, coronal and sagittal, from left to right. The center of all the images is located at (10, 10, 5).

Fig. 7.

Estimated coefficient images obtained from four classification methods in Simulation II. The top panels are the respective coefficients from EN-LR and GN-LR, and the bottom panels are the coefficients from SMAC-I and SMAC-II. The first two coefficient images (β_1 and β_2) of each classifier are displayed. The coefficients from SMAC-I and SMAC-II are obtained using Equation (6). All the coefficient images are displayed in the transverse direction, centered at (16, 16, 1).

Fig. 8.

Classification results under different training sample sizes. The left panel displays the classification accuracy and right panel displays the area under the ROC curve.

Fig. 9. Classification results under different noise levels. The left panel displays classification accuracy and the right panel displays the area under the ROC curve.

Fig. 10.

Estimated coefficient images obtained from five classification methods in the binary ADNI study. The five plots are the respective coefficient images from EN-LR, GN-LR, SSVM, SMAC-I and SMAC-II. Each coefficient is displayed in the views of coronal, sagittal and transverse planes. The slices are located at (0, -17, 18).

Fig. 11.

Estimated coefficient images obtained from four classification methods in the multi-category ADNI study. The four rows of plots are the respective coefficient images from EN-LR, GN-LR, SMAC-I and SMAC-II. The first two coefficient images (β_1 and β_2) of each classifier are displayed. The coefficients from SMAC-I and SMAC-II are obtained using Equation (6). Each coefficient is displayed in the views of coronal, sagittal and transverse planes. The slices are located at $(0, -17, 18)$.

Fig. 12.

Mean computational time for each method in Simulation I. In each plot, the vertical direction represents the value of λ_1 , from top to bottom being $\{0, 2^{-14}, 2^{-13}, \dots, 2^5\}$, and the horizontal direction represents λ_2 , from left to right being $\{0, 2^{-14}, 2^{-13}, \dots, 2^5\}$.

Table 1

List of parameters in Algorithm 2.4.2.

Parameter(s)	Description
β	Target variable in the optimization.
$\mathbf{v}_1$	Auxiliary variable, $\mathbf{v}_1 = \mathbf{A}\beta$.
$\mathbf{v}_2$	Auxiliary variable, $\mathbf{v}_2 = \mathbf{R}\beta$.
$\mathbf{v}_3$	Auxiliary variable, $\mathbf{v}_3 = \tilde{\mathbf{B}}\beta$.
$\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3$	Dual variables in the ADMM.
λ_1, λ_2	Penalty strength for L_1 and TV-I/II respectively.
$\mathbf{A}$	Matrix to compute functional margin, see equation (10).
$\mathbf{K}^t$	Vector to be calculated for solving (14), $\mathbf{K}^t = \mathbf{A}^T(\mathbf{v}_1^t - \mathbf{u}_1^t) + \mathbf{R}^T(\mathbf{v}_2^t - \mathbf{u}_2^t)$.
$\tilde{\mathbf{B}}$	Augmented discrete operator for FL penalty, see Section 2.4.1 for details.
$\mathbf{M}$	Selection matrix to rule out additional terms, see Section 2.4.1 for details.
$\mathbf{R}$	Recovering matrix for masked images.
Γ_1	The first column of matrix $\mathbf{I} + \tilde{\mathbf{B}}^T\tilde{\mathbf{B}}$.

Table 2

Demographic Information of All Subjects in our Data Analysis. The unit for intracranial volume (ICV) is $10^3 \times \text{cm}^3$; for age and ICV, the means are reported, with standard deviations in parentheses.

	Male	Female	Age	ICV
NC	111	95	76.03 (4.95)	1.27 (0.12)
MCI	233	131	75.00 (7.38)	1.29 (0.14)
AD	98	81	75.50 (7.53)	1.27 (0.15)

Table 3

Comparison of Results in the Classification of Binary Synthetic Data. Classification accuracy (ACC), true positive rate (TPR), true negative rate (TNR) and area under the ROC curve (AUC); values are reported as percentages; means from 50 iterations are reported, with standard deviations in parentheses.

Method	ACC	TPR	TNR	AUC
EN-LR	73.75 (5.24)	76.44 (7.00)	71.05 (8.50)	81.70 (5.99)
GN-LR	93.04 (1.75)	93.01 (1.94)	93.07 (1.92)	98.11 (0.79)
SSVM	92.23 (2.06)	94.11 (2.88)	90.35 (3.37)	97.91 (1.00)
SMAC-I	96.52 (1.21)	95.83 (2.16)	97.21 (1.30)	99.51 (0.31)
SMAC-II	96.17 (1.32)	95.54 (1.98)	96.79 (1.72)	99.39 (0.36)

Table 4

Comparison of Results in the Classification of Multi-category Synthetic Data. Classification accuracy (ACC); area under the ROC curve (AUC1-3) values for classes 1, 2 and 3, respectively; values are reported as percentages; means from 50 iterations are reported, with standard deviations in parentheses.

Method	ACC	AUC1	AUC2	AUC3
EN-LR	64.89 (3.29)	63.96 (2.70)	86.41 (3.02)	86.54 (3.22)
GN-LR	92.17 (1.13)	90.74 (4.60)	99.55 (0.16)	99.23 (0.26)
SMAC-I	94.70 (1.10)	95.65 (1.42)	99.85 (0.06)	99.66 (0.13)
SMAC-II	94.19 (0.86)	95.85 (1.12)	99.71 (0.07)	99.64 (0.10)

Table 5

Sample Size Sensitivity Analysis. Columns are different training sample sizes; classification accuracy (ACC); area under the ROC curve (AUC); values are reported as percentages; evaluation is based on 600 test samples.

Method	10	20	30	40	50	60	70	80	90	100
ACC										
EN-LR	54.17	61.67	60.17	68.83	72.67	73.50	78.33	82.00	84.50	83.50
GN-LR	65.17	78.00	90.17	91.50	91.83	91.00	94.83	89.83	95.67	96.67
SSVM	59.33	82.50	92.67	93.50	93.50	90.67	93.67	94.83	94.67	93.67
SMAC-I	65.00	80.17	95.50	96.50	97.00	95.00	96.50	97.00	97.67	96.50
SMAC-II	55.00	89.67	96.83	96.17	96.83	96.00	91.00	95.50	95.67	97.33
AUC										
EN-LR	56.69	65.97	64.51	78.84	81.12	79.63	85.88	90.72	92.77	92.92
GN-LR	71.45	86.44	96.59	97.43	97.23	97.39	98.91	97.13	99.09	99.48
SSVM	62.89	91.34	97.77	98.66	98.32	96.42	98.33	98.70	98.81	98.33
SMAC-I	77.50	90.15	99.42	99.65	99.73	98.98	99.71	99.69	99.76	99.63
SMAC-II	59.76	96.68	99.57	99.67	99.58	99.14	97.37	99.42	99.24	99.69

Table 6

Noise Level Sensitivity Analysis. Columns are different standard deviations of noise; classification accuracy (ACC); area under the ROC curve (AUC); values are reported as percentages; evaluation is based on 600 test samples.

Method	1.0	1.25	1.5	1.75	2.0	2.25	2.5	2.75	3.0	3.25	3.5	3.75	4.0
ACC													
EN-LR	97.00	94.83	84.33	78.50	73.50	72.50	64.50	64.17	58.67	60.67	54.17	58.17	57.17
GN-LR	100.00	99.00	98.67	97.17	91.00	88.00	81.00	77.50	72.00	69.17	66.67	63.17	61.83
SSVM	99.67	98.83	96.50	93.50	90.67	84.83	80.50	76.33	72.50	68.83	65.33	57.00	56.00
SMAC-I	98.50	98.83	98.17	96.17	95.00	93.50	90.83	88.00	83.00	71.67	67.50	65.83	58.00
SMAC-II	99.33	98.67	97.00	98.00	96.00	94.33	89.33	80.83	76.17	82.00	69.17	69.33	70.83
AUC													
EN-LR	99.97	98.81	92.90	87.89	79.63	80.64	69.03	67.56	61.13	63.61	54.55	60.72	58.42
GN-LR	100.00	99.96	99.91	99.65	97.39	95.42	89.43	85.46	80.61	76.38	73.31	69.15	67.66
SSVM	99.98	99.97	99.59	98.20	96.42	92.83	88.57	83.96	79.15	74.78	70.52	59.82	58.46
SMAC-I	99.84	99.94	99.87	99.31	98.98	97.86	96.75	95.19	91.30	78.25	74.06	73.44	62.56
SMAC-II	100.00	100.00	99.82	99.87	99.14	98.69	95.37	88.76	85.01	90.72	77.93	78.52	78.13

Table 7

Comparison of Results in the Binary Classification of MRI Data. Classification accuracy (ACC); area under the ROC curve (AUC); values are reported as percentages; means from 30 iterations are reported, with standard deviations in parentheses.

Method	NC vs AD		MCI vs AD		NC vs MCI	
	ACC	AUC	ACC	AUC	ACC	AUC
EN-LR	86.84 (3.13)	93.33 (1.90)	69.22 (2.65)	66.31 (4.90)	70.35 (2.07)	73.33 (2.41)
GN-LR	86.23 (2.78)	95.95 (1.36)	65.67 (3.23)	68.15 (5.39)	68.55 (2.29)	76.69 (2.99)
SSVM	86.14 (3.41)	92.65 (3.02)	48.07 (5.47)	56.35 (4.21)	63.72 (0.00)	50.00 (0.00)
SMAC-I	89.12 (2.30)	93.92 (1.20)	69.13 (2.72)	67.24 (4.62)	70.68 (2.43)	75.31 (2.22)
SMAC-II	88.33 (3.32)	93.97 (1.44)	69.19 (3.59)	66.86 (4.64)	70.38 (2.51)	76.35 (2.48)

Table 8

Comparison of Results for Simultaneously Classifying 3 Categories of MRI Data. Classification accuracy (ACC); area under the ROC curve (AUC1-3) with respective reference labels NC, MCI and AD; values are reported in percentages; means from 30 iterations are reported, with standard deviations in parentheses.

Method	ACC	AUC1	AUC2	AUC3
EN-LR	49.32 (3.18)	72.39 (3.15)	50.68 (4.66)	77.56 (4.44)
GN-LR	49.75 (2.55)	77.90 (2.99)	51.94 (4.83)	82.89 (1.45)
SMAC-I	53.22 (2.90)	81.01 (3.23)	50.31 (5.03)	77.53 (3.63)
SMAC-II	52.68 (4.20)	81.75 (3.85)	48.85 (4.79)	78.21 (3.67)