Journal of Neuroscience Methods 282 (2017) 81–94

Contents lists available at ScienceDirect

Journal of Neuroscience Methods

journal homepage: www.elsevier.com/locate/jneumeth

Decoding the encoding of functional brain networks: An fMRI classification comparison of non-negative matrix factorization (NMF), independent component analysis (ICA), and sparse coding algorithms

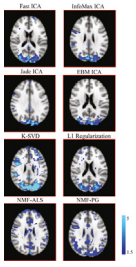
Jianwen Xie[a], Pamela K. Douglas[b], Ying Nian Wu[a], Arthur L. Brody[b], Ariana E. Anderson[a,b,*]

[a] Department of Statistics, University of California, Los Angeles, United States
[b] Department of Psychiatry and Biobehavioral Sciences, University of California, Los Angeles, United States

HIGHLIGHTS

- We compared ICA, K-SVD, NMF, and L1-Regularized Learning for encoding brain components within an fMRI scan.
- The temporal weights of each encoding were used to predict activity using machine learning classifiers.
- NMF, which eliminates negative BOLD signal, performed poorly compared to ICA and sparse coding algorithms (K-SVD, L1 Regularized Learning).
- L1 Regularized Learning and K-SVD frequently outperformed four variations of ICA to predict fMRI task activity.
- Spatial sparsity of encoding maps were associated with increased classification accuracy, holding constant effects of algorithms.

GRAPHICAL ABSTRACT



Visual network manually identified across each algorithm, within a single scan. Sparsifying algorithms (K-SVD and LASSO/L1-Regularization) outperformed ICA and NMF algorithms for predicting whether a subject was viewing a video, listening to an audio stimulus, or resting, during an fMRI scan. Maps were rescaled to be on common scale for illustration purposes.

ARTICLE INFO

Article history:
Received 6 November 2016
Received in revised form 7 March 2017
Accepted 7 March 2017
Available online 18 March 2017

Keywords:
FMRI
Classification
ICA
NMF
K-SVD
L1 Regularized Learning
Independent component analysis

ABSTRACT

Background: Brain networks in fMRI are typically identified using spatial independent component analysis (ICA), yet other mathematical constraints provide alternate biologically-plausible frameworks for generating brain networks. Non-negative matrix factorization (NMF) would suppress negative BOLD signal by enforcing positivity. Spatial sparse coding algorithms (L1 Regularized Learning and K-SVD) would impose local specialization and a discouragement of multitasking, where the total observed activity in a single voxel originates from a restricted number of possible brain networks.
New method: The assumptions of independence, positivity, and sparsity to encode task-related brain networks are compared; the resulting brain networks within scan for different constraints are used as basis functions to encode observed functional activity. These encodings are then decoded using machine learning, by using the time series weights to predict within scan whether a subject is viewing a video, listening to an audio cue, or at rest, in 304 fMRI scans from 51 subjects.
Results and comparison with existing method: The sparse coding algorithm of L1 Regularized Learning outperformed 4 variations of ICA ($p < 0.001$) for predicting the task being performed within each scan

* Corresponding author at: Department of Psychiatry and Biobehavioral Sciences, University of California, Los Angeles, 760 Westwood Plaza, Los Angeles, CA 90095, United States.
E-mail address: arianaanderson@mednet.ucla.edu (A.E. Anderson).

http://dx.doi.org/10.1016/j.jneumeth.2017.03.008

Non-negative matrix factorization
Machine learning
Random forests
Support vector machines
Artifacts
Negative BOLD signal
Sparsity
Image processing
Pattern recognition

using artifact-cleaned components. The NMF algorithms, which suppressed negative BOLD signal, had the poorest accuracy compared to the ICA and sparse coding algorithms. Holding constant the effect of the extraction algorithm, encodings using sparser spatial networks (containing more zero-valued voxels) had higher classification accuracy ($p < 0.001$). Lower classification accuracy occurred when the extracted spatial maps contained more CSF regions ($p < 0.001$).
Conclusion: The success of sparse coding algorithms suggests that algorithms which enforce sparsity, discourage multitasking, and promote local specialization may capture better the underlying source processes than those which allow inexhaustible local processes such as ICA. Negative BOLD signal may capture task-related activations.

1. Introduction

Although functional MRI (fMRI) data contain numerous spatio-temporal observations, these data putatively reflect changes in a relatively small number of functional networks in the brain. These underlying processes can be modeled using any number of blind source separation (BSS) algorithms, with independent component analysis (ICA) being the most commonly used to extract hypothesized brain networks. Other mathematical constraints, such as positivity or sparsity, provide alternative interpretations for how the brain generates and encodes functional brain components. Comparing how BSS algorithms capture task-related activations will simultaneously evaluate different interpretations of functional encoding; decoding the fMRI encodings provides a "hypothesis test" for this purpose (Naselaris et al., 2011).

The observed fMRI volume at a given time instance can be modeled as a linear combination of weighted spatial components X according to $y = dX$. This linear model encodes an fMRI scan, where the spatial maps correspond to a brain network, and the time series weights describe the contribution of that spatial map to the total functional activity observed at a given time. This linear model can be computed using a variety of BSS methods including: ICA, K-Means Singular Value Decomposition (K-SVD), $L1$ Regularized Learning, and non-negative matrix factorization (NMF). Each of these methods applies different constraints to recover and numerically unmix sources of activity, leading to different representations of functional brain components (spatial maps) and time series weights. These components are often interpreted as brain networks.

ICA finds components that are maximally statistically independent in the spatial or temporal domain. Since the number of time points is typically less than the number of voxels, spatial ICA in particular has become the most studied approach for extracting and summarizing brain activity components in fMRI (McKeown et al., 1997), where many of the components computed by ICA correspond to previously identified large scale brain components (Smith et al., 2009). In spatial ICA, maximizing spatial independence suggests that the ability of a voxel to contribute to a given brain network is not contingent on how it contributes to any other brain network, or to how many other brain networks it contributes; there is no bound to how strongly a voxel can contribute to a network since arbitrarily large positive and negative "activations" are permitted across components (Calhoun et al., 2004). This permits local activations to be inexhaustible; no region is disqualified from participating in any brain network simply because it already participates strongly in another network. The validity of the spatial and temporal independence assumptions in fMRI have been the subject of lively debate for some time (Friston, 1998; McKeown et al., 2003; Daubechies et al., 2009; Calhoun et al., 2013). The most frequently-used ICA algorithms maximize independence by maximizing the non-Gaussianity (Fast ICA) (Hyvärinen and Oja, 1997) or minimizing the mutual information (InfoMax) (Bell and Sejnowski, 1995). Recently ProDenICA has been shown to perform better on resting-state fMRI data (Risk et al., 2013).

K-SVD (sparse dictionary learning) has been used to nominate components both on the voxel (Lee and Ye, 2010; Lee et al., 2011; Abolghasemi et al., 2013) and region of interest (ROI) scale (Eavani et al., 2012), and restricts many of the voxels in a network to have null (zero) values by limiting component membership of each voxel. This biologically plausible constraint prohibits multitasking of a voxel, as no voxel can contribute to all processes simultaneously (Spratling, 2014). Specifically, because a voxel must contribute by magnitude zero to many components, it is permitted to contribute to only a limited number of remaining components by the mathematical constraint itself. More recently it was applied in simulated functional connectivity (FC) analyses to recover true, underlying FC patterns (Leonardi et al., 2014), where dynamic FC was found to be better described during periods of task by alternating sparser FC states. Similarly, $L1$ Regularized Learning (Lasso) was applied in Parkinson's Disease to analyze fMRI functional connectivity during resting state (Liu et al., 2015). Non-negative matrix factorization has been applied to fMRI data (Wang et al., 2004; Potluru and Calhoun, 2008; Ferdowsi et al., 2010, 2011), where the alternating least squares NMF algorithm has been found superior to detect task-related activation compared to three other NMF algorithms (Ding et al., 2012). Imposing non-negativity in NMF suppresses all negative BOLD signal, while the parts-based representation which results from this non-negativity suggests that a subset of local-circuits may be a better representation of functional activity than geographically-distributed components. Previously, we used NMF to identify multimodal MRI, fMRI, and phenotypic profiles of Attention Deficit Hyperactivity Disorder (ADHD) (Anderson et al., 2013). Although the biological interpretations of these algorithms are not mutually exclusive to ICA, the success of a specific dictionary learning method may prioritize theories of encoding. Moreover, given the finding that the choice of regularizer may be more important than the choice of classifier (Churchill et al., 2014), judicious selection *a priori* of a regularization method may allow us to better understand the network dynamics of cognitive processes.

In this paper, we evaluate which of the individual subject representations computed by ICA, K-SVD, NMF, and $L1$ Regularized Learning best encode task-related activations that are pertinent for classification, by embedding these components as features within a decoding framework. These representations can be compared by using the time series weights of the spatial maps for task classification, where each time point in an fMRI scan is encoded using the functional brain components proposed by each algorithm. Using the time series weights for the encodings, we predict which task a subject was performing during a scan by employing support vector machines (SVM) (Burges, 1998) and random forests (Breiman, 2001). We have recently leveraged component feature weights for classification of fMRI data in a number of studies (Douglas et al., 2009, 2011, 2013; Anderson et al., 2010, 2012). We compare the predictive accuracies of the algorithms while varying the number of components and the presence of artifactual components (effects of motion, non-neuronal physiology, scanner artifacts and other noisy sources). Finally, we evaluate how physiological profiles of the proposed brain components (tissue activation densities and

sparsity) are associated with the classification accuracy, to compare whether the algorithms are substantially different after accounting for the physiological profiles of the spatial components they extract. Collectively, this paper evaluates the performance of different algorithms in encoding functional brain components, possible correlates for explaining their performance, and the assumptions they support.

2. Materials and methods

2.1. Overview

We will compare the representations of each algorithm by using the time series weights to predict, within a single scan, which activity a subject was doing. We evaluate not only the general algorithms, but also their varied implementations, including four variations of ICA (Entropy Bound Minimization [EBM ICA], Fast ICA, InfoMax ICA, Joint Approximate Diagonalization of Eigen-matrices [JADE ICA]), two variations of NMF (Alternating Least Squares [NMF-ALS], Projected Gradient [NMF-PG]), and two sparse coding algorithms ($L1$ Regularized Learning, K-SVD). Finally, we assess how the physiological profiles of the spatial maps may correlate with the ability to encode an fMRI scan, holding constant the effect of the algorithm.

2.2. Data: design, experiment, preprocessing and cleaning

We describe briefly the experimental design and experiment here; it is discussed in detail in Culbertson et al. (2011). A total of 51 subjects were scanned in a study on craving and addiction. The subjects were divided into three groups, and scanned up to 3 times before and after treatment (with bupropion, placebo, or counseling) while watching a video and receiving audio cues meant to induce nicotine cravings, in a blocked design task. All scans contained all stimuli in a blocked design. This led to a total of 304 usable scans, after removing 2 scans for which scan-time was abbreviated. There were a total of 18 nicotine related video cues, and 9 neutral video cues. The audio cues were reportedly difficult to hear at times due to scanner noise. These volunteers were scanned using a gradient-echo, echo planar imaging sequence with a TR of 2.5 s, echo time, 45 ms; flip angle, 80; image matrix, 128 64; field of view, 40.20 cm; and in-plane resolution, 3.125 mm.

The fMRI data processing was carried out using FEAT (FMRI Expert Analysis Tool) version 6.00, part of FSL (FMRIB's Software Library, https://www.fmrib.ox.ac.uk/fsl). The following preprocessing was applied in routine order; motion correction using MCFLIRT (Jenkinson et al., 2002); non-brain removal using BET (Smith, 2002); spatial smoothing using a Gaussian kernel of FWHM 5 mm; grand-mean intensity normalisation of the entire 4D dataset by a single multiplicative factor; highpass temporal filtering (Gaussian-weighted least-squares straight line fitting, with sigma = 50.0 s). The BSS algorithms were performed within the subjects' native space (not aligned to MNI) because the prediction was performed and cross-validated within each scan, and not across scans.

Preprocessing steps in fMRI data arguably are a regularization method, since preprocessing algorithms apply numerous transformation matrices through filtering, alignment, and removal of physiological artifacts (Churchill et al., 2015). Because of this, we secondarily investigated the role of additional processing on the classification accuracy, since the Fast ICA algorithms was used to identify artifacts. We evaluated the classification accuracy of these algorithms on scans containing two levels of noise: the first dataset was traditionally-preprocessed using the default FSL pipeline, while the second dataset consisted of the same set of scans which had been additionally cleaned using the FIX artifact removal tool (Griffanti et al., 2014; Salimi-Khorshidi et al., 2014), where approximately 50% of components (defined by ICA) were flagged as possible noise (residual effects of motion, non-neuronal physiology, scanner artifacts and other noisy sources) and removed. Following the removal of the possible artifacts, each scan was reconstructed, and the dictionary learning algorithms were rerun within each "cleaned" scan.

2.3. BSS algorithms

K-SVD, NMF, ICA, and $L1$ Regularized Learning all perform a matrix factorization into spatial maps (components) and time series weights. They differ primarily however in what constraints they impose when learning, and whether the primary emphasis is on learning the spatial or temporal features. For example, NMF places equal emphasis on learning the time series weights and the components, while spatial ICA imposes statistical independence over space (but no constraints over time). In addition, the sparse coding algorithms considered here emphasize learning the time series weights instead of the spatial maps, and imposes spatial sparsity within voxel across components by restricting the total contribution of a given voxel over all components. Because all the algorithms assessed here produce both spatial maps and time courses, this distinction is not restrictive in comparing the ability of spatial maps to summarize functional activation patterns at a given time point. We uniformly describe the algorithms in the context of $Y = DX$, where Y is the original data matrix (a single fMRI scan) of size $n \times m$ containing m voxels and n time points, D is the mixing matrix containing the time series weights for k components of dimension $n \times k$ where $k \leq n$, and X is the matrix of spatial maps (components) of dimension $k \times m$. We extract both $k \in 20, 50$ components within each scan for ICA, NMF, K-SVD and $L1$ Regularized Learning.

2.3.1. Spatial statistical independence: independent component analysis

Statistical independence is an important concept that constitutes the foundation of ICA, which is one of the most widely used blind source separation techniques for exposing hidden factors underlying sets of signals (Aapo Hyvärinen, 2000; Aapo Hyvärinen and Juha Karhunen, 2001). ICA can be written as a decomposition of a data matrix $Y_{n\times m}$ into a product of (maximally) statistically independent spatial maps (components) $X_{k\times m}$ with a mixing matrix (time series weights) $D_{n\times k}$, given by $Y_{n\times m} = D_{n\times k}X_{k\times m}$. Formally, spatial independence implies that for a given voxel m, $p(x_{1m}, x_{2m}, \ldots, x_{km}) = \prod_{i=1}^{k} p(x_{1m})p(x_{2m}), \ldots, p(x_{km})$ Here, we examine ICA with a constraint of (maximal) spatial independence, using the Fast ICA, InfoMax ICA, EBM ICA, and JADE ICA algorithms.

Different measurements of independence govern different forms of the ICA algorithms, resulting in slightly different unmixing matrices. Minimization of mutual information and maximization of non-Gaussianity are two broadest measurements of statistical independence for ICA. Fast ICA (Hyvärinen, 1999) is a fixed point ICA algorithm that maximizes non-Gaussianity as a measure of statistical independence, motivated by the central limit theorem. Fast ICA measures non-Gaussianity by negentropy, which itself is the difference in entropy between a given distribution and the Gaussian distribution with the same mean and variance. InfoMax ICA (Bell and Sejnowski, 1995) belongs to the minimization of mutual information family of ICA algorithms; these find independent signals by maximizing entropy. Instead of directly estimating the

entropy, the EBM ICA algorithm (Li and Adali, 2010) approximates the entropy by bounding the entropy of estimates using numerical computation. Due to the flexibility of the entropy bound estimator, EBM ICA can be applied to data that come from different types of distributions, e.g., sub- or super-Gaussian, unimodal or multimodal, symmetric or skewed probability density functions, placing an even stronger emphasis on independence. JADE (Cardoso, 1999; Cardoso and Souloumiac, 1993) is an ICA algorithm exploiting the Jacobi technique to perform joint approximate diagonalization on fourth-order cumulant matrices to separate the source signals from mixed signals. Typical ICA algorithms use centering, whitening, and dimensionality reduction as preprocessing steps. Whitening and dimension reduction can be achieved with principal component analysis (PCA) or Singular Value Decomposition (SVD). They are simple and efficient operations that significantly reduce the computational complexity of ICA, so are applied in the implementations of ICA here.

2.3.2. Positivity: non-negative matrix factorization

Psychological and physiological evidence show that component parts play an important role in neural encoding of the visual appearance of an object in the brain (Palmer, 1977; Logothetis and Sheinberg, 1996; Wachsmuth et al., 1994). When applied to fMRI data, this parts-based representation is conceptually similar to encouraging neighboring voxels to co-activate, which would encourage spatial smoothness in the resulting component maps. The non-negativity in NMF constraints would eliminate negative BOLD signal changes which have been controversially associated with sources such as cerebral blood volume changes and inhibition (Bianciardi et al., 2011; Harel et al., 2002; Moraschi et al., 2012; Smith et al., 2004).

Non-negativity is a useful constraint for matrix factorization which leads to parts-based representation, because it allows only additive (positive) combinations of the learned bases (Lee and Seung, 1999). Given a $n \times m$ non-negative data matrix Y and a prespecified $k < \min(n, m)$, NMF finds two non-negative matrices $D_{n\times k}$ and $X_{k\times m}$ such that $Y \approx DX$. The conventional approach to find D and X is by minimizing the squared error between Y and DX:

$$\min_{D,X}\{\|Y - DX\|_2^2\} \text{ subject to } D_{ia} \geq 0, X_{bj} \geq 0, \forall i, a, b, j \quad (1)$$

which is a standard bound-constrained optimization problem. There are several algorithms in which the D and X may be found. The most commonly known approach is the multiplicative update method (Lee and Seung, 2001). NMF-ALS has previously shown stronger performance than multiplicative update-NMF in fMRI (Ding et al., 2012).

2.3.2.1. NMF-ALS. A more flexible and general framework for obtaining NMF is to use alternating least squares (ALS), which was first introduced by Paatero (1994) in the middle of the 1990s under the name positive matrix factorization (Anttila et al., 1995; Paatero, 1994). The ALS algorithm does not have the restriction of locking 0 elements in matrices D and X. The framework of ALS is summarized as follows:

(1) Initialize $D_{ia}^0 > 0, X_{bj}^0 > 0, \forall i, a, b, j$.

(2) For $J = 0, 1, 2, \ldots$

$$D^{(J+1)} = \arg\min_{D\geq 0}\|Y - DX^{(J)}\|_2^2 \quad (2)$$

$$X^{(J+1)} = \arg\min_{X\geq 0}\|Y - D^{(J+1)}X\|_2^2 \quad (3)$$

The iterations can be performed with an initialization of D and X, and then alternating between solving (2) and (3) until a stopping criterion is satisfied. ALS is also known as "block coordinate descent" approach in bound-constrained optimization (Bertsekas, 1999). We refer to (2) or (3) as a sub-problem in this. At each step of iterations, finding an optimal solution of the nonnegative least squares sub-problem is important because otherwise, the convergence of overall algorithm may not be guaranteed (Kim and Park, 2007).

Some successful NMF algorithms are based on ALS; their differences arise from using different ways to solve the ALS sub-problems. As an elementary strategy, the alternating least squares algorithm (Paatero, 1994; Berry et al., 2007) solves the sub-problems by an unconstrained least squares solution (without the nonnegativity constraint), *i.e.*, $D \leftarrow ((XX^T)^{-1}XY^T)^T$ and $X \leftarrow (D^TD)^{-1}D^TY$, and every negative element resulting from least squares computation is set to zero to ensure nonnegativity after each update step. The implementation for ALS described above is very fast, and requires less work than other NMF algorithms; however, setting negative elements to 0 in order to enforce nonnegativity is quite *ad hoc*.

2.3.2.2. NMF-PG. Alternating nonnegative least squares using projected gradients (NMF-PG) has been used for NMF (Lin, 2007). The sub-problems in ALS above are solved here using projected gradient methods. To calculate X, the algorithm updates it by the rule $X \leftarrow P[X - \alpha \nabla f(X)]$, where $P[\cdot]$ is a bounding function or projection operator that maps a point back to the bounded feasible region, $\nabla f(X)$ is the gradient function computed as $D^T(DX - Y)$, and α is the step size. Selecting the step size α for each sub-iteration in NMF-PG is a main computational task. The same approach is utilized to calculate D.

2.3.3. Sparse coding: K-SVD and L1 Regularized Learning

All dimension reduction methods necessarily provide compression, and some variations of independence also encourage sparsity (*e.g.* the InfoMax variant of ICA). Similarly, the non-negativity constraint in NMF shrinks the intensity of many spatial maps' voxels to zero which also encourages sparsity. However, the sparsity obtained in these methods is a secondary benefit to the primary intention (independence and non-negativity), and not the primary objective of the algorithms. We thus describe the "sparse coding" algorithms not on whether they may encourage sparsity, but rather on whether they enforce it. Sparse coding in fMRI restricts a single voxel to have a limited number of (approximately) non-null values across components. This sparsity on the voxel scale across spatial maps necessarily provides sparsity within each spatial map as well.

2.3.3.1. K-SVD. K-SVD is a generalization of the k-means algorithm; in the k-means algorithm, an observation can be represented by its centroid (the central point of the cluster to which that element belongs). In K-SVD, an observation is instead modeled as a weighted combination of multiple (but not all) dictionary elements-effectively imposing the $L0$-norm on how many components a specific voxel can participate in. K-SVD is a sparse data-representation algorithm to learn an over-complete dictionary, such that any single observation is constructed from a subset of the total number of dictionary elements.

Sparsity is enforced over space by limiting the number of elements which can be used to construct that observation, and the dictionary elements D learned are the corresponding time series weights; when applied to fMRI, this constraint more generally suggests a local specialization; a single voxel can only contribute to a subset of all ongoing processes. The weights of the dictionary (time series weights) are the spatial maps themselves. This bypasses the need for the PCA which typically precedes ICA. The sparse coding constraint over space suggests that the components are best represented by a subset of all voxels; this is in direct contrast to algorithms such as ICA, where every voxel is allowed to contribute, in varying degrees, to the representative time series weights.

K-SVD operates by iteratively alternating between (1) sparse coding of the data based on the current dictionary (estimating the spatial maps, when applied to fMRI), and (2) dictionary updating (revising the time series weights) to better fit the observed data (Aharon and Bruckstein, 2006; Rubinstein et al., 2010). A full description of the K-SVD algorithm is given as follows:

Task: Find the best dictionary to represent the data samples $\{y_i\}$ as sparse compositions, by solving

$$\min_{D,X}\{\|Y - DX\|_2^2\}\text{subject to}\forall i,\ \|x_i\|_0 \le T_o \quad (4)$$

Initialization: Set the dictionary matrix $D^{(0)} \in R^{n\times K}$ with l^2 normalized columns. Set $J = 0$, the counting index. Let n = the number time points, m= the number of voxels, and let K = the number of dictionary elements being estimated.

Main Iteration: Increment J by 1, and apply:

Sparse Coding Stage: Use any pursuit algorithm to compute the representation vectors x_i for each example y_i, by approximating the solution of

$$i = 1, 2, \quad ldots, m, \min_{x_i}\{\|y_i - D^{(J-1)}x_i\|_2^2\}\text{subject to}\|x_i\|_0 \le T_0 \quad (5)$$

Dictionary Update Stage: for each column $k = 1, 2, \ldots, K$ in $D^{(J-1)}$, update it as follows:

(1) Define the group of observations that use this atom, $\omega_k = \{i \mid 1 \le i \le m, x_T^k(i) \ne 0\}$.
(2) Compute the overall representation error matrix, E_k, by $E_k = Y - \sum_{j\ne k} d_j x_T^j$.
(3) Restrict E_k by choosing only the columns corresponding to ω_k, and obtain E_k^R.
(4) Apply SVD decomposition $E_k^R = U\Delta V^T$. Choose the updated dictionary column d_k to be the first column of U. Update the coefficient vector x_R^k to be the first column of V multiplied by $\Delta(1, 1)$.

Stopping rule: If the change in $\|Y - D^{(J)}X^{(J)}\|_2^2$ is small enough, stop. Otherwise, iterate further.

Output: The desired results are dictionary D^J and encoding X^J.

Due to the $L0$-norm constraint, seeking an appropriate dictionary for the data is a non-convex problem, so K-SVD does not guarantee to find the global optimum (Rubinstein et al., 2010).

L1 Regularized Learning

We can relax the $L0$-norm constraint over the coefficients x_i by instead using a $L1$-norm regularization (Olshausen et al., 1996), which enforce x_i ($i = 1, \ldots m$) to have a small number of nonzero elements. Then, the optimization problem can be written as:

$$\min_{D,X}\{\|Y - DX\|_2^2\} + \beta\sum_i \|x_i\|_1\text{subject to}\|d_j\|_2 \le 1, \forall j = 1, 2, \ldots, k \quad (6)$$

where a unit $L2$-norm constraint on d_j typically is applied to avoid trivial solutions.

Due to the use of $L1$ penalty as the sparsity function, the optimization problem is convex in D (while fixing X) and convex in X (while fixing D), but not convex in both simultaneously. Lee et al. (2007) optimizes the above objective iteratively by alternatingly optimizing with respect to D (dictionary) and X (coefficients) while fixing the other. For learning the coefficients X, the optimization problem can be solved by fixing D and optimizing over each coefficient x_i individually:

$$\min_{x_i}\{\|y_i - Dx_i\|_2^2\} + \beta\|x_i\|_1 \quad (7)$$

which is equivalent to $L1$-regularized least squares problem, also known as the Lasso in statistical literature. For learning the dictionary D, the problem reduces to a least square problem with quadratic constraints:

$$\min_D\|Y - DX\|_2^2\text{subject to}\|d_j\|_2 \le 1, \forall j = 1, 2, \ldots, k \quad (8)$$

2.4. Implementation details: BSS algorithms

Here, we have tried to explore common variations for each learning algorithm as thoroughly as is computationally feasible. Full implementation code is provided in the Data in Brief, with a summary of implementation provided here. This section describes the implementation and the crucial parameters used in each learning algorithm. Given an input matrix Y, all the algorithms were initialized by randomly generating matrix D and X, and run a sufficiently large number of iterations to obtain the converged results. For most of the algorithms, the number of iterations we used is 400, upon which we verified convergence using appropriate fit indices.

NMF: We used Matlab's embedded function *nnmf* for NMF-ALS and (Lin, 2007) for NMF-PG in our experiment. Maximum number of 400 iterations is allowed. InfoMax ICA: We used the EEGLAB toolbox (Delorme and Makeig, 2004) for InfoMax ICA, which implements logistic InfoMax ICA algorithm of Bell and Sejnowski (Bell and Sejnowski, 1995) with the natural gradient feature (Amari et al., 1996), and with PCA dimension reduction. Annealing based on weight changes is used to automate the separation process. The algorithm stops training if weights change below 10^{-6} or after 500 iterations.

Fast ICA: We used the Fast ICA package (the ICA and BSS group, U.o.H., 2014), which implements the fast fixed-point algorithm for ICA and projection pursuit. PCA dimension reduction and hyperbolic tangent for nonlinearity are used.

JADE ICA: We used the Matlab implementation of JADE (Cardoso and Souloumiac, 1993) for the ICA of real-valued data. PCA is used for dimension reduction before the JADE algorithm is performed.

EBM ICA: We used the Matlab implementation of EBM ICA (Li and Adali, 2010) for real-valued data. Four nonlinearities (measuring functions) x^4, $|x|/(1+|x|)$, $x|x|/(10+|x|)$, and $x/(1+x^2)$ are used for entropy bound calculation. This implementation adopts a two-stage procedure, where the orthogonal version of EBM ICA, with measuring function x^4 and maximum number of iterations of 100, is firstly used to provide an initial guess, and then the general nonorthogonal EBM ICA with all measuring functions uses the linear search algorithm to estimate the demixing matrix. The technique for detection and removal of saddle convergence proposed in Koldovsky et al. (2006) is used in orthogonal EBM ICA if the algorithm converges to a saddle point. Similar to other ICA methods, PCA for dimension reduction was used before the algorithm is performed.

K-SVD: While the total number of components was allowed to vary (either 20 or 50), each voxel was allowed to participate in only $K = 8$ components. This corresponds to 40% of all components for the 20-component extractions, and 16% of components for the 50-component extraction. The K-SVD-Box package (Rubinstein and Elad, 2008) was used to perform K-SVD, which reduces both the computing complexity and the memory requirements by using a modified dictionary update step that replaces the explicit SVD computation with a much quicker approximation. It employs the Batch-OMP (orthogonal matching pursuit) to accelerate the sparse-coding step. Implementation details can be found in Rubinstein and Elad (2008).

$L1$ Regularized Learning: We used the implementation of efficient sparse coding proposed by Lee et al. (2007). It solves the $L1$-regularized least squares problem iteratively by using the feature-sign search algorithm and $L2$-constrained least squares problem by its Lagrange dual. The parameter for sparsity regularization is 0.15 and for smoothing regularization is 10^{-5}.

2.5. Implementation details: machine learning algorithms and parameters

All scans contained all three activities in a blocked design: video, audio, and rest. Half of the 212 timepoints were captured during rest, while the remaining timepoints were divided equally between blocks of video or audio stimuli. Using the time series weights for each algorithm extracted within each scan, we predicted which of three activities a subject was performing using both an SVM classifier (using a 10-fold cross-validation Varoquaux et al., 2016) as well as a random forests classifier (which provides the out-of-bag testing error). In separate analyses, we also tested whether the results found here were consistent when predicting other binary variants of the stimuli, such as rest vs. activity (video, audio), video vs. no-video, and audio vs. no-audio.

This classification procedure was done separately for each machine learning algorithm using both 20 and 50 component extractions, and for both the traditionally-preprocessed and artifact-suppressed data, to assess the impact of both component number and the effect of residual noise. The most stringent data cleaning involved cleaning the scans twice, using traditional preprocessing and FIX where artifactual components were identified and discarded from the scan (residual effects of motion, non-neuronal physiology, scanner artifacts and other noisy sources). For example, for 20 network time-series weights extracted using NMF-ALS for a single scan within an iteration of 10-fold cross-validation, we trained an SVM model using 90% of the 212 time-series weights to predict whether a subject was viewing a video, listening to an audio cue, or resting on the remaining 21 time-points. All available time-series weights (either 20 or 50) were used. Using the time series weights for prediction is similar to projecting the entire fMRI scan onto the spatial components defined by the algorithms. The average classification accuracy over 304 scans measures the predictive performance of each algorithm, for the specified number of components and artifact suppression level.

SVM with a radial basis kernel was implemented within R (Meyer et al., 2012), with results presented using default parameter settings (cost parameter: 1, gamma: 0.05). We present the results for the untuned algorithm to avoid introducing bias into the cross-validation, but also test whether parameter tuning differentially affects the performance of the SVM algorithm for specific BSS methods, by comparing also the accuracy when varying the cost parameter and gamma for each algorithm. For multiclass-classification with 3 levels as was implemented here (video, audio, rest), libsvm (called from R) uses the "one-against-one" approach, in which 3 binary classifiers are trained; the appropriate class is found by a voting scheme. Random Forests was implemented within R with 500 trees using default parameter settings (Liaw and Wiener, 2002). Within each node, $floor(\sqrt{n})$ features were randomly selected and chosen to partition the features; for the 20 components, 4 variables were tried at each split. For the 50 components, 7 variables were tried.

2.6. Measuring noise and measuring sparsity within extracted components

After performing two iterations of data cleaning (traditional preprocessing and FIX artifact removal), we subsequently measured whether residual noise and sparsity may impact the classification accuracy. We hypothesized that "activation", or high-intensity voxel values, within CSF regions may be an indicator of the overall level of noise within a spatial map. For the spatial maps created by running each BSS algorithm on the artifact-cleaned scans, we measured the average intensity of voxels within CSF, grey, and white matter regions. The T1 MNI tissue partial volume effect (PVEs) were aligned into the subject's functional space via the subject's T2 structural scan in a two-step process. First, the segmented MNI images were aligned into the subject's structural space using the whole-brain MNI-152 to the subject's T2 mapping learned using FLIRT. Then, we registered these PVE images into the subject's functional MRI space using the subject's T2 to fMRI mapping.

Using these tissue masks we computed the average intensity of the extracted spatial maps within regions probabilistically defined as grey matter, white matter, and CSF. This was computed using the cross-correlation of each tissue-type partial volume effect (PVE) with each functional map; for a given algorithm, the 20 components extracted for a scan would yield 60 correlation measures with the grey, white, and CSF maps. The average and the variation of the tissue types in the 20 components were used to summarize the overall distribution of "active" voxels in the spatial maps. These tissue-region correlates were computed for the 2432 basis sets extracted for all algorithms.

To measure sparsity for each BSS algorithm within each scan, we computed the *L*0-norm of each spatial map, and used the average across all components within a scan to measure the spatial sparsity of the extracted components. Specifically, for each spatial map we measured sparsity using the *L*0-norm, where $sparsity(X) = \frac{-1}{k} * \sum_{j=1}^{k} \|X_j\|_0$ where k is the total number of components. The negative sign ensures that more zero-valued voxels will lead to a lower sparsity measure. This *L*0 based measure was chosen because it is not sensitive to the scaling of the images, which are necessarily different across algorithms.

2.7. Comparison of BSS algorithms by classification accuracy

The BSS algorithms' SVM performance were first compared for the 20 component, artifact-cleaned scans, predicting the untuned classification accuracy using Algorithm as a main effect in a general linear mixed-effects regression model (baseline model); Scan-ID and Subject-ID were included as random effects to adjust for multiple comparisons. The random effects account for the covariance that is present within subject and within scan, as we would expect, for example, a subject to show similar components across multiple scans. Similarly, a specific scan being evaluated using ICA and NMF would show more similarity in the resulting components than two different scans being evaluated separately with ICA and NMF. Including "Algorithm" as a main effect allows estimation of the effect of the specific algorithm, after holding constant the effects of the scan, subject, and session. Across the 304 scans and 8 algorithms, this resulted in the classification accuracy from 2432 untuned SVM models being compared. We then assessed whether sparser spatial maps led to better classification accuracy after adjusting for the effect of the algorithm, predicting classification accuracy using both the BSS algorithm type and component sparsity as fixed effects and Scan-ID and Subject-ID as random effects for multiple comparison adjustments.

Finally, we included tissue-type profiles to measure the association of the physiological profiles of the spatial maps with their ability to decode functional activity. This assesses, among other things, whether component extractions containing larger amounts of white matter have poorer classification accuracy than component extractions containing more grey matter. In a general linear mixed effects regression model (full model), we predicted the classification accuracy within each scan using the algorithm type, a session effect, the sparsity of the components, the average and standard deviation of the intensity within regions of grey matter, white matter, CSF (averaged across components). Scan ID and Subject ID were both included as random effects to adjust for multiple comparisons. This was done for the 20 component extraction, on data which had been cleaned of artifacts (including white matter and CSF artifacts).

Table 1
Classification accuracy averaged across 304 traditionally preprocessed data scans in predicting whether a subject was viewing a video, listening to an audio stimuli, or resting, using 20 components. Chance accuracy is 50%.

	SVM	Random Forests
NMF-ALS	0.63 (0.08)	0.63 (0.09)
NMF-PG	0.63 (0.08)	0.63 (0.09)
InfoMax ICA	0.64 (0.08)	0.67 (0.09)
JADE ICA	0.66 (0.08)	0.69 (0.09)
EBM ICA	0.66 (0.10)	0.71 (0.10)
Fast ICA	0.67 (0.08)	0.72 (0.08)
K-SVD	0.70 (0.08)	0.73 (0.08)
*L*1 Regularized Learning	0.74 (0.07)	0.71 (0.07)

Table 2
Classification accuracy averaged across 304 traditionally preprocessed scans in predicting whether a subject was viewing a video, listening to an audio stimuli, or resting, using 50 components. Chance accuracy is 50%.

	SVM	Random Forests
InfoMax	0.59 (0.07)	0.59 (0.07)
NMF-ALS	0.60 (0.08)	0.62 (0.10)
NMF-PG	0.66 (0.09)	0.64 (0.08)
JADE ICA	0.70 (0.08)	0.72 (0.08)
EBM ICA	0.72 (0.09)	0.74 (0.09)
Fast ICA	0.73 (0.07)	0.75 (0.07)
K-SVD	0.75 (0.07)	0.77 (0.07)
*L*1 Regularized Learning	0.75 (0.07)	0.69 (0.07)

We compared the baseline model containing just the BSS algorithm main effect to the full model containing the BSS algorithm effect and the physiological profiles of the spatial maps using a chi-square hierarchical regression, to evaluate whether the physiological profiles of the spatial maps were related to classification accuracy above and beyond the effect of the BSS algorithm alone.

3. Results

All algorithms performed better than chance on average, with general trends present across the constraint families. The best performing independence algorithm (Fast ICA) was still inferior to the worst performing sparse coding algorithm (K-SVD) for classifying cognitive activity ($p < 0.005$) in Table 1 and Fig. 1, using 20 dictionary elements on the traditionally-preprocessed fMRI data and an SVM classifier. The strong predictive performance of the sparse coding algorithms persisted when using 50 components instead of 20 (Table 2) and when using data from which additional artifacts had been removed, shown in Table 3, although there were some excpetions.

We compared how similarly these algorithms classified within each scan in Fig. 2, using a multi-dimensional scaling of the accuracies within scan (for the 20-component, traditionally preprocessed scans). Methods predicted similarly to other methods within their class: K-SVD, a sparse coding method, was most correlated with *L*1 Regularized Learning (Lasso), another sparse coding method. This suggests that algorithms within certain families tend to have similar performance.

Within each scan, the algorithms in order from best to worst classification accuracy were: *L*1 Regularization, K-SVD, Fast ICA, EBM ICA, JADE ICA, InfoMax ICA, NMF-PG, and NMF-ALS, as shown

Table 3
Classification accuracy averaged across 304 artifact-cleaned scans in predicting whether a subject was viewing a video, listening to an audio stimuli, or resting, using 20 components. Chance accuracy is 50%.

	SVM	Random Forests
NMF-ALS	0.58 (0.09)	0.60 (0.11)
NMF-PG	0.58 (0.09)	0.59 (0.11)
InfoMax ICA	0.67 (0.07)	0.67 (0.07)
JADE ICA	0.68 (0.08)	0.71(0.08)
EBM ICA	0.69 (0.09)	0.72 (0.09)
Fast ICA	0.70 (0.08)	0.72 (0.08)
K-SVD	0.70 (0.08)	0.71 (0.08)
*L*1 Regularized Learning	0.74 (0.07)	0.71 (0.07)

in Table 4 using 20 spatial maps on artifact-cleaned data. Spatial sparsity was highly significant; scans which extracted sparser spatial maps had higher classification accuracy ($p < 0.001$) after accounting for the effect of the algorithms as shown in Table 5. The CSF functional map density was negatively associated with classification accuracy ($p < 0.001$), while high variability in white matter functional density was associated with better classification accuracy ($p < 0.001$). Including the physiological profiles of the spatial maps significantly helped predict the within-scan classification accuracy, above and beyond the effect of the algorithm alone ($p < 0.001$, chi-square test of nested models). The physiological profiles of the spatial maps were likely correlated with sparsity of the spatial maps, as the sparsity measurement lost significance once accounting for the physiological profiles (Table 6).

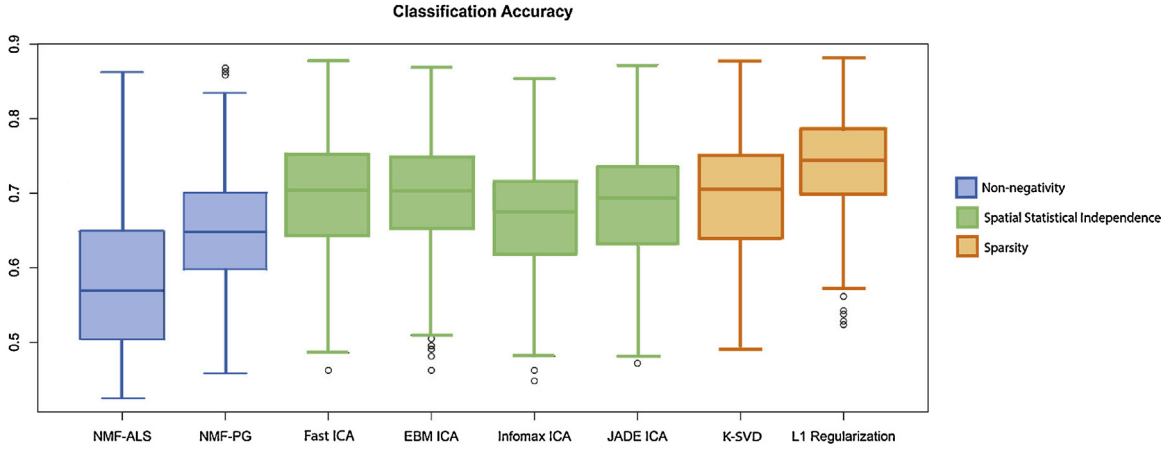


Fig. 1. Although ICA is the most commonly used method to extract and define fMRI spatial components, encoding scans using sparse coding algorithms like K-SVD and *L*1 Regularized Learning led to higher classification accuracy in determining whether a subject was resting, watching a video, or hearing an audio cue. The best performing independence algorithm (Fast ICA) was still inferior to the worst performing sparse coding algorithm (K-SVD) for classifying cognitive activity ($p < 0.005$) using an SVM classifier on 20 components on traditionally-preprocessed data. Chance accuracy is 50%.

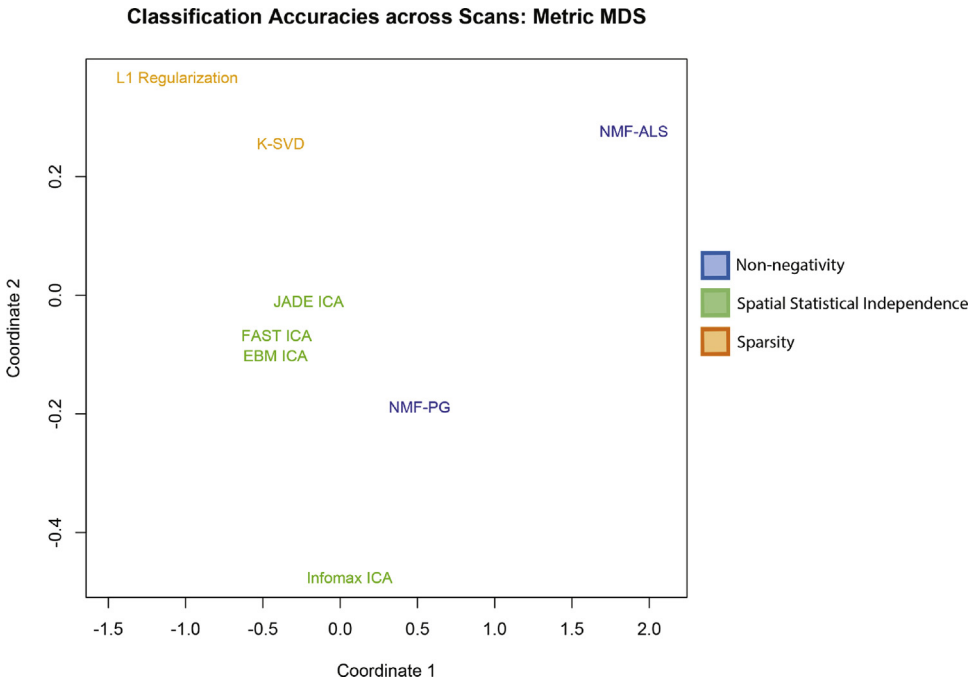


Fig. 2. Within a scan, algorithms sharing particular constraints tended to classify with similar accuracy. This multi-dimensional scaling visualizes the similarities of how the BSS algorithms classified within each scan using the 20-basis components extracted on the traditionally preprocessed data. Each dimension represents a composite of features that are relevant to explaining the covariance structure of the classification accuracy, similar to the dimensions extracted in a traditional PCA.

Table 4
Classification accuracy of BSS algorithm compared to Fast ICA, using 20 components extracted from artifact-cleaned scans, in order of performance from worst to best. Time series weights from InfoMax ICA, JADE ICA, NMF-PG, NMF-ALS predicted activity significantly worse than Fast ICA, while L1-Regularization did significantly better ($p < 0.001$). Baseline is set to Fast ICA untuned SVM classification accuracy. Scan-ID and Subject-ID were included as random effects within a general linear mixed-effects regression model to adjust for multiple comparisons.

	Estimate	Std. error	*t* value
(Intercept)	0.70	0.01	110.55***
NMF-ALS	−0.11	0.001	−37.84***
NMF-PG	−0.05	0.001	−15.59***
InfoMax ICA	−0.03	0.001	−10.11***
JADE ICA	−0.01	0.001	−4.02***
EBM ICA	−0.001	0.001	−0.91
K-SVD	0.001	0.001	0.36
*L*1 Regularized Learning	0.04	0.001	14.70 ***

*** $p < .001$.

Table 5
Greater sparsity for an extracted spatial map was associated with a higher classification accuracy in predicting a subject's task during scan time when using those spatial maps for encoding ($p < 0.001$), holding constant the effect of the algorithm. Using 20 components extracted from artifact-cleaned scans, sparsity was measured using the negative averaged number of zero-valued voxels of all spatial maps, which is insensitive to the scaling of the individual algorithms. Baseline is set to Fast ICA untuned SVM classification accuracy. Scan-ID and Subject-ID were included as random effects within a general linear mixed-effects regression model to adjust for multiple comparisons.

	Estimate	Std. error	*t* value
(Intercept)	0.83	0.01	57.91***
Sparsity	0.001	0.001	10.82***
NMF-ALS	−0.14	0.001	−35.21***
K-SVD	−0.08	0.01	−9.99***
NMF-PG	−0.05	0.001	−17.48***
InfoMax ICA	−0.03	0.001	−10.39***
JADE ICA	−0.01	0.001	−4.13***
EBM ICA	−0.001	0.001	−0.93
*L*1 Regularized Learning	0.03	0.001	10.70**

** $p < .01$.
*** $p < .001$.

Table 6
Encodings using spatial maps with high intensity in CSF regions had reduced classification accuracy, while spatial maps with variable extractions in white-matter and grey-matter regions had higher classification accuracy. Baseline is set to Fast ICA untuned SVM classification accuracy. Scan-ID and Subject-ID were included as random effects within a general linear mixed-effects regression model to adjust for multiple comparisons.

	Estimate	Std. error	*t* value
(Intercept)	0.71	0.02	30.53***
Mean(CSF)	−0.30	0.09	−3.34***
Mean(Grey Matter)	−0.05	0.08	−0.65
Mean(White Matter)	0.01	0.06	0.08
SD(CSF)	0.001	0.11	0.01
SD(Grey Matter)	−0.32	0.09	−3.62***
SD(White Matter)	0.44	0.07	6.07***
Sparsity	0.001	0.001	0.68
Session	−0.01	0.01	−1.57
NMF-ALS	−0.05	0.02	−2.23
InfoMax ICA	−0.03	0.001	−9.69***
EBM ICA	−0.01	0.01	−1.66
JADE ICA	−0.01	0.001	−3.54***
K-SVD	0.01	0.01	0.54
NMF-PG	0.07	0.02	3.25***
*L*1 Regularized Learning	0.07	0.001	13.99***

*** $p < .001$.

4. Discussion

Our experiments showed that algorithms which enforced sparsity, instead of merely encouraging it, frequently had the highest classification accuracy compared to the independence and sparse coding algorithms. When we implemented K-SVD with an overcomplete basis of 300 components, the classification accuracy remained similar. The non-negativity algorithms had the least accurate classification accuracy. Among the ICA algorithms, Fast ICA had the highest classification accuracy. These trends were consistent for the extraction of 20 and 50 components, and for different levels of data cleaning (removing artifactual components). Although our presented results are for untuned SVM (cost parameter: 1, gamma: 0.05), we demonstrated that tuning these

parameters did not change the relative performance of each algorithm. For all algorithms, extracted spatial maps containing more regions of CSF led to worse classification accuracy ($p < 0.001$). CSF and white matter artifacts were purposefully removed during the artifact cleaning, so this CSF measures the residual noise. Algorithms which showed large variability in their extraction of white matter regions had significantly higher classification accuracy ($p < 0.001$). This may suggest that purposefully selecting white matter regions to construct functional components helps to improve classification accuracy.

The BSS algorithms can be formulated as theories of functional organization and encoding. Comparing the classification accuracy can neither validate nor invalidate any specific hypothesis, but the success of a specific BSS algorithm provides evidence for a unique biological interpretation of generative activity. Interpreting features in any decoding scheme is complex (Guyon and Elisseeff, 2003). As recently highlighted by Haufe et al. (2014), feature weights should not be interpreted directly. When features have a shared covariance structure, irrelevant features can be assigned a strong weight to compensate for shared noise in feature space. However, as these authors also point out, when the features are independent, the shared covariance is minimized and the interpretation theoretically becomes more tractable.

Over-sparsification of brain components to optimize classification accuracy can lead to brain components that omit important brain regions (Rasmussen et al., 2012). The impact of even small brain lesions on cognitive and motor processes is inarguable, demonstrating that every voxel has an important role. The resulting spatial (sub)components produced by sparsity algorithms are the foundation stones for the full underlying processes which occur during cognition. Because of this, they may be most interpretable in conjunction with other sparse components. When sparsity algorithms are implemented on a voxel-wise basis for classification, they remove those voxels which are highly correlated with other predictive voxels. In the context of fMRI brain components, the sparsity is not applied to the classification algorithm itself or across voxels, but rather within a single voxel. Because of this, correlated processes and components may be consolidated. In the context of this dataset, subjects were viewing cues related to nicotine, so nicotine related cues may not be seen as a separate network, as they would likely be correlated with the activity of viewing a video.

Although sparsifying algorithms may be optimal for predicting large effects such as visual stimuli, the drawbacks to such algorithms may be the collapsing or consolidation of more nuanced activations. Sparsifying algorithms may be acting as a feature selector that avoids the noise inherent in using an fMRI hemodynamic response as a proxy for a neuronal activation. Because the actual neuronal activity associated with a task may be blurred by hemodynamic filters, a voxel may not be identified as a significant predictor of an activity even when its contribution may still be moderate. The hemodynamic response has previously been shown to impact fMRI classification of a video stimulus, similar to our work here on predicting video activation Mandelkow et al. (2016). The success of sparsitfying algorithms realized here in fMRI may not be seen in more direct imaging modalities such as EEG, which do not pass through a hemodynamic filter.

In Fig. 3, we show thresholded spatial maps derived from the visual task for a representative subject. These maps were all rescaled to match across algorithms since, for example, NMF values are all positive while ICA may take on any value range. Most of the algorithms clearly isolated the visual network. However, the NMF algorithms resulted in weakly identifiable visual components, and unsurprisingly this class of algorithms were outperformed by the other algorithms. It is therefore reasonable to suggest that the signed values in the BOLD signal carry either important

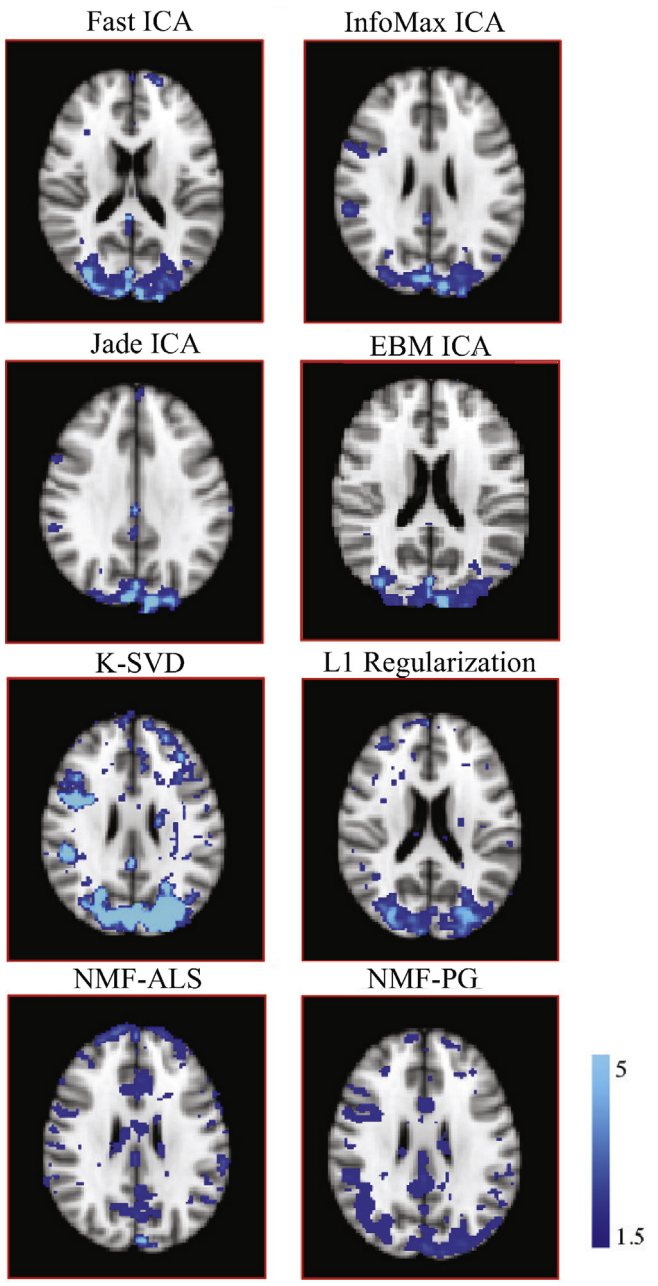


Fig. 3. Visual network spatial maps for each of the compared algorithms identified via manual inspection out of 20 possible components extracted within a single scan. The sparsity algorithms limited component membership across time – this permits a given network to capture a process in a single spatial network, but prohibits it from distributing the activity in a given voxel across networks. Because of this, a sparsity algorithm would prohibit multiple visual networks across components, by consolidating activity into limited number of networks. The amount of spatial sparseness for a specific image map is ultimately dependent upon how the data are rescaled and thresholded. The raw maps are provided in the appendix to allow unthresholded observations.

descriptive or class specific information. In addition to identifying possible components through manual inspection of the spatial maps, we also identified them by the time course. The unthresholded spatial map most correlated (absolute value) with performing any task vs. rest is shown in Fig. 4. This may be the most likely candidate for the default mode network since the signs of the timecourses are arbitrarily positive or negative, and the "any task" timecourse consisted of both auditory and visual stimuli which are functionally distinct – thus precluding this as a specific task-related

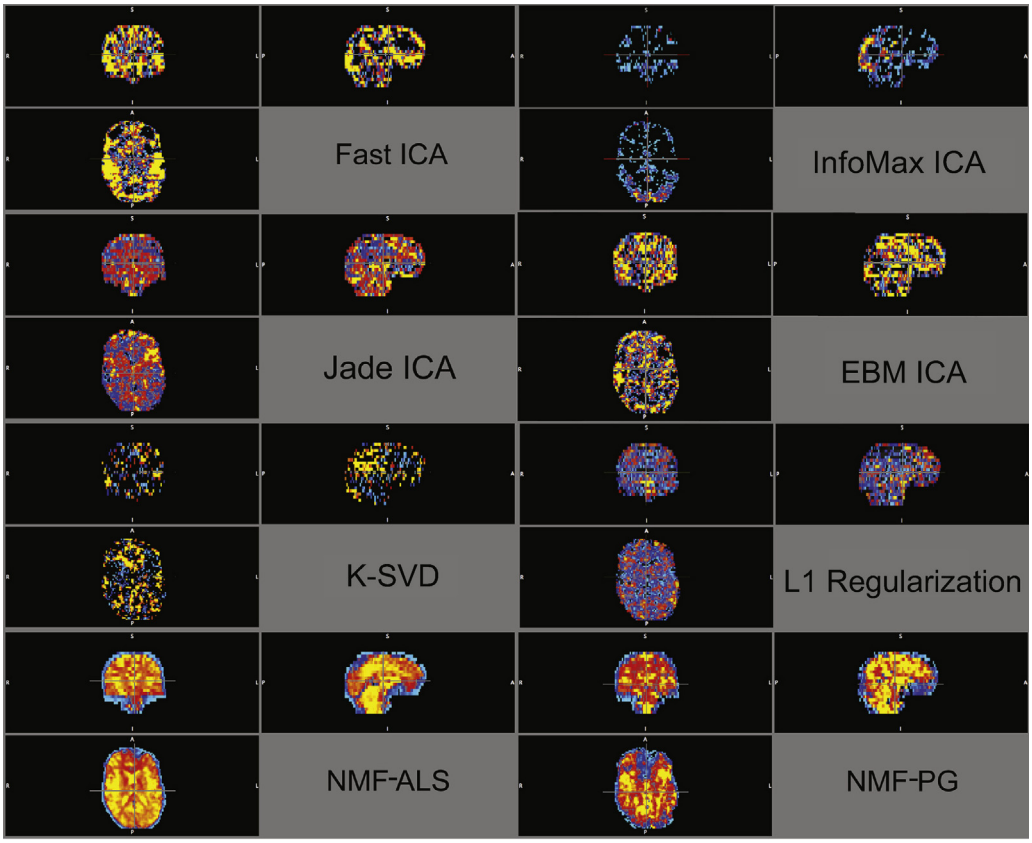


Fig. 4. Candidate Default Mode Network. This component was identified as having the temporal activity most correlated with the any-task vs. rest pattern, and may correspond to the default mode network. The maximally correlated component for each algorithm is shown unthresholded, but consistently colored within each algorithm. High intensity regions are red, and low intensity regions are blue. The sparsity of these raw networks, which haven't been rescaled and thresholded commonly across algorithms, are in contrast with the rescaled visual network presented above. The units of the color scale depend on the algorithms; for example, NMF values would be bounded below by 0, while the intensity of the ICA spatial maps were largely centered between (−3, 3). Algorithms were performed to extract 20 dictionary elements within a single scan on traditionally preprocessed data. Raw data are provided in Appendix, including the functional and structural scan to enable readers to evaluate these encodings. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.)

network. High-intensity values for that algorithm are in red, while low-intensity values are in blue, but the numeric values do not map across algorithms because of the different rescaling methods these algorithms employ. The differences among algorithms is partially attributed to the physiological profiles of the spatial maps they extracted. The 20 extracted components for all algorithms for a single scan are provided in the Data in Brief as NIfTI files, along with code to reproduce these results.

Despite the promising implications of NMF, its predictive performance was underwhelming. The non-negativity constraint suppresses all negative BOLD activations, which has controversial hypotheses behind its generation. The poor performance of NMF may suggest that the disregarded negative activations did contain task-related signal. The non-negativity constraint also encourages local circuits, and geographically distributed components may have been parsed and diluted by this. The NMF-PG algorithm actually performed better than Fast ICA after adjusting for the CSF functional loadings and the white matter variability. This suggests that NMF algorithms may capture higher levels of noise in the fMRI activation maps. The NMF spatial maps had a strong resemblance to structural images with the regional intensity varying depending on the tissue type, even though the learning was done on functional data. This was not an artifact of the initialization, as the maps were initialized to be random values. It was also not an artifact of improper convergence, as running the algorithm far beyond convergence did not change the structural-map appearance of these maps. Rather, we speculate that the likely default mode may have been suppressed in NMF, since the DMN is anticorrelated with the task related components which may have been stronger contenders spanning both auditory and visual domains. It is possible that NMF performed poorly because it was not able to use activation of the DMN to identify the periods of rest. Finally, the NMF-ALS algorithm had difficulty extracting a full set of 20 components on the artifact-cleaned fMRI data, although it was able to do so consistently on the fMRI data which had received only regular pre-processing. Collectively, this suggests that NMF may find its purpose not in capturing task-related activation maps, because of its suppression of all negative signals.

Our interrogation into these components is based upon our perturbation of the system using a stimulus, where we use task decoding to identify properties of components which should exist based on the visual and audio stimuli present. The full presented results predict any task (rest, video, or audio) within a single scan-however, we also tested models which classified states specifically as rest vs. activity, video vs. no-video (audio, rest), and audio vs. no-audio (video, rest), and realized similar results to those presented here. This suggests that our results are stable across the stimuli tested here. Out of the algorithms considered here, the sparse coding assumptions may be more biologically plausible in accounting for task-related activity changes, but restricts inference only to components which are task-associated. Moreover, this does not suggest that the frameworks imposed by these specific algorithms are the best explanation for how functional components are organized; rather, these analyses suggest that out of the linear

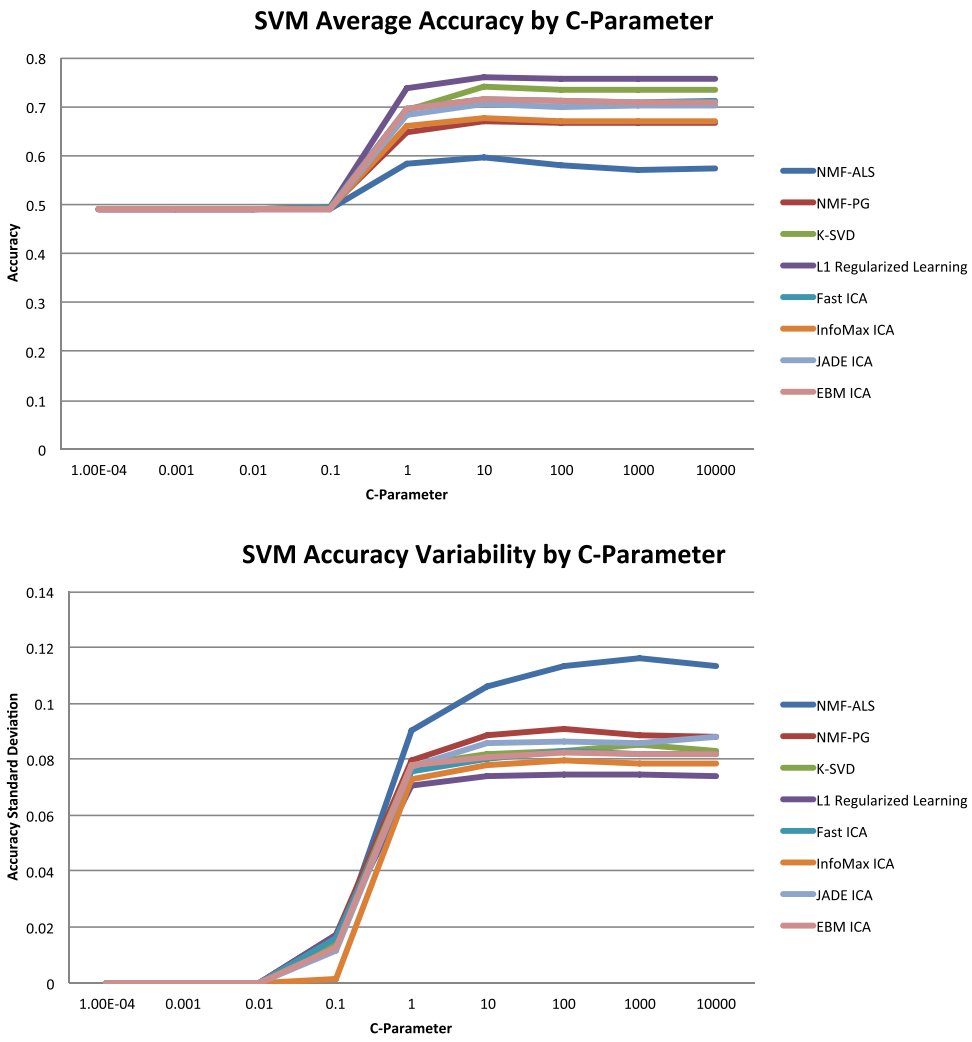


Fig. 5. Varying the C-parameter affected the overall classification accuracy within each algorithm, but the relative performance across algorithms remained unchanged. Algorithms with greater classification accuracy had reduced variability in accuracy across possible choices of the C-parameter. Classification accuracy averaged across 304 artifact-cleaned scans in predicting whether a subject was viewing a video, listening to an audio stimuli, or resting, using 20 components. Chance accuracy is 50%.

algorithms defined by specific constraints, the sparse coding constraint may best capture task-related activations.

There are several limitations to this paper. We assessed how brain components are encoded using a decoding approach-however, this is an indirect and a secondary measurement of performance. It is arguably unknown how many brain components exist in a given subject, and whether these brain components are consistent across subjects – thus our extracted components may be combining multiple brain components. For the classification, given the computational considerations when optimizing many parameters, we used a fixed 'C' penalty term for analysis presented here – however, for the secondarily-cleaned extraction of 20 components, we also performed a secondary investigation into how the 'C' penalty affected the performance of the various algorithms. Our results suggested that although there was accuracy to be gained by optimizing the C-parameter, the effects of this parameter were uniform across algorithms, as shown in Fig. 5. Similarly, we saw little interaction between the choice of the gamma parameter and the accuracy of the different algorithms, as shown in Fig. 6.

Although we used 304 scans for this study, these scans originated from 51 subjects. Although we controlled for the effects of Subject, Scan, and Session, there may have been other unknown factors which introduced covariance. The role of smoothing done in preprocessing may also be influential, as well as any unknown parametric variations in software such as FSL that were not similarly performed when implementing this in Matlab; however, we compared subsets of these analyses with and without smoothing, and also with an FSL implementation of FAST-ICA in Melodic, and reached very similar findings.

The thresholding of components was set similarly across algorithms, but 20 components in ICA may capture a different profile of information than 20 components extracted using NMF. However, because we were also interested in secondary measures (e.g. variability of physiological tissue profiles), permitting the number of components to vary across algorithms, or even within algorithms, would have reduced inference on our ability to interpret the role of different tissue types on sparsity and classification accuracy. We found also that comparing instead the classification accuracies across different algorithms while not holding constant the number of components still yields consistent findings the best performing FAST-ICA still underperformed the worst-performing *L*1 Regularization for SVM classification across all combinations of component numbers, machine-learning classifiers, and data cleaning levels, providing reassurance that holding constant parameters across algorithms may not be introducing substantial bias.

We cleaned the data (motion correction, etc...) in two stages, through traditional pre-processing and secondarily through artifact-removal using FIX. The secondary artifact-cleaning step

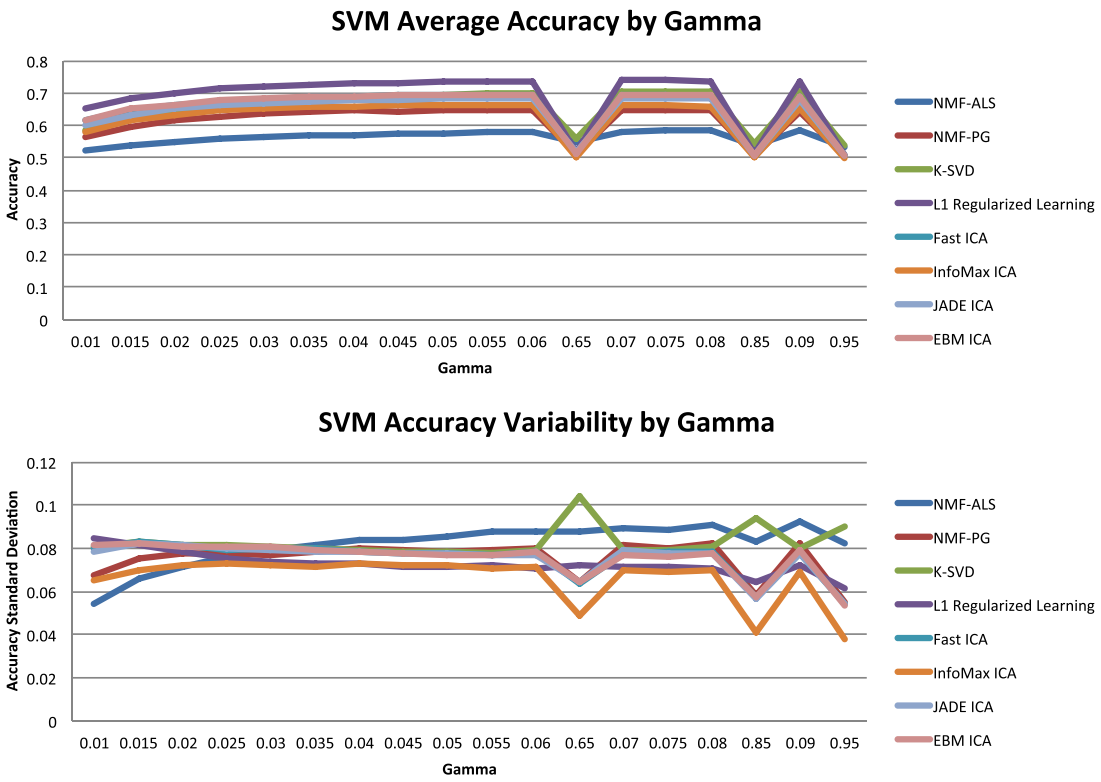


Fig. 6. Similar to the effect of the C-parameter, varying the gamma-parameter did not differentially impact the classification accuracy for any specific algorithm. Classification accuracy averaged across 304 artifact-cleaned scans in predicting whether a subject was viewing a video, listening to an audio stimuli, or resting, using 20 components. Chance accuracy is 50%.

used Fast ICA to identify artifactual components, which were flagged and removed before reconstructing the scans. The ICA-based artifact cleaning may have been a disadvantage for other algorithms, since these algorithms were run on ICA-cleaned data. When implementing the additional Fast ICA based artifact removal, the Fast ICA features performed 1% better than the sparsity algorithms when using random forests, but not when using SVM as shown in Table 3. When optimizing the SVM C-parameter, at its maximum accuracy the Fast ICA features still performed 3% worse than the K-SVD features, and 5% worse than the *L*1 Regularized Learning features. Although the mechanism for this is unclear it does suggest that the classifier itself may interact with the regularization methods. The two stages can be optimized both together and separately.

Despite secondary cleaning using artifact-removal, some residual noise may still have remained, as demonstrated by the CSF intensity in the cleaned data predicting poor classification accuracy within a scan. Although we referred to the spatial maps as brain components here, the variability of these components for the different algorithms gives pause as to whether these components are really the linear combinations as the BSS algorithms assume to produce functional activity. We did not evaluate every existing variation of ICA, NMF, and other sparse coding algorithms, but relied instead on the most popular variations. Similarly, these analyses only compared within-scan component extractions, and may not generalize to group analyses where components are extracted across large numbers of subjects and scans. It is possible that the sparse coding algorithms may be hypertuned to the unique activity within a single scan, and that the components observed when applied to groups may not be flexible enough to account for the variation seen across subjects. Investigating how these components change in group analyses, and whether sparsifying methods are still superior on a group scale, are both directions for future research.

5. Conclusion

The more common ICA-based methods of interpreting functional brain components were, on average, suboptimal for creating a decoding basis set. More generally, these results suggest that the functional organization of the brain may be modeled better by sparse coding algorithms than by spatial independence or parts-based representations, at the lowest levels. We argue these results are reasonable, where a sparse operational framework would support an efficient use of biological resources. This does not preclude a network-based view of functional activity rather, it emphasizes that processes are purposefully allocated to specific components, and that components themselves are specialized. The SVM algorithm, at its heart, uses weighted combinations of sparse elements to predict which activity a subject was performing. This suggests that the sparse interpretation allows a more flexible construction of the comprehensive components than components which by their mathematical assumptions nurture more fully-defined functional maps. Differences between the SVM and random forest classifier even when using the same features suggests an interplay between the regularization and the classification itself; there was an interaction between the classification algorithm for decoding and the BSS algorithm for encoding, with the *L*1 Regularization algorithm in particular performing between 3% and 6% poorer when using random forests compared to SVM. This suggests a fundamental dependency between encoding and decoding, where their performance is yoked.

We suggest that sparsity is a reasonable developmental constraint when applied across processes, since redundant (non-sparse) network components are costly in energy consumption and complexity. Sparse coding schemes reduce energy consumption by minimizing the number of action potentials necessary to represent information within neural codes (Spratling, 2014; Sengupta

et al., 2010). Moreover, the specialization and discouragement of local multitasking associated with spatial sparsity may be more reasonable than the "boundless energy" framework of statistical independence, where a single voxel is not constrained by its overall (total) component membership. The non-negativity NMF algorithms performed poorly, suggesting that the negative BOLD signals are indeed important measures of task-related functional activity. In ICA, a single voxel's intensity in a given component is not influenced by the separate cognitive processes (components) to which that voxel contributes, permitting unlimited multitasking. The success of the sparse coding algorithms suggests that, during periods of activity, local activity is specialized.

Sparse algorithms may capture more efficient representations for classification because of the elimination of redundant features, which leads to a parsimonious framework. The power of sparse algorithms may be in their limited disbursement of functional activity across components; the components they then nominate hold power when combined with other sparse components, suggesting they form a metabasis for cognitive activity rather than complete components. This may indicate that these sparse elements will prove more useful for hierarchical algorithms such as deep learning methods. This is a direction for future research. The experiments presented here suggest the importance of sparsity, but indeed sparsity may not be the key factor in decoding fMRI activity. When including the anatomical profiles of the spatial components, the extraction of CSF regions (a proxy measure of overall noise) overtook sparsity in explanatory power to predict classification accuracy. This may suggest then that sparsity is a proxy measure for reduced noise, but not necessarily the driving source behind the predictive power. The sparser maps may be more successful across all algorithms because they correspond well to the actual biological processes, because they omit regions containing noise, or even because the sparser basis sets span the partitioning space better. Although sparsity presents an optimal encoding framework for applying decoding models, the mechanism underlying its success remains unknown.

Acknowledgments

Our sincere appreciation to M.S. Cohen and M.A.O. Vasilescu for constructive feedback and insight on this manuscript.

Ariana E. Anderson, Ph.D., holds a Career Award at the Scientific Interface from BWF and is supported by R03 MH106922 from NIMH and K25 AG051782 from NIA. Pamela K. Douglas is supported by Klingenstein Third Generation Fellowship, Keck Foundation, and NIH/National Center for Advancing Translational Science (NCATS) UCLA CTSI Grant Number UL1TR000124. Ying Nian Wu is supported by NSF DMS 1310391, ONR MURI N00014-10-1-0933, DARPA MSEE FA8650-11-1-7149. Arthur L. Brody is supported by the National Institute on Drug Abuse (R01 DA20872), the Department of Veterans Affairs, Office of Research and Development (CSRD Merit Review Award I01 CX000412), and the Tobacco-Related Disease Research Program (23XT-0002).

References

Aapo Hyvärinen, E.O., 2000. Independent component analysis: algorithms and application. Neural Netw. 13 (4–5), 411–430.
Aapo Hyvärinen, Juha Karhunen, E.O., 2001. Independent Component Analysis. Wiley, New York.
Abolghasemi, V., Ferdowsi, S., Sanei, S., 2013. Fast and incoherent dictionary learning algorithms with application to fMRI. Signal Image Video Process., 1–12.
Amari, S.i., Cichocki, A., Yang, H.H., et al., 1996. A new learning algorithm for blind signal separation. Adv. Neural Inf. Process. Syst., 757–763.
Anderson, A., Dinov, I.D., Sherin, J.E., Quintana, J., Yuille, A.L., Cohen, M.S., 2010. Classification of spatially unaligned fMRI scans. NeuroImage 49 (3), 2509–2519.
Anderson, A., Douglas, P.K., Kerr, W.T., Haynes, V.S., Yuille, A.L., Xie, J., Wu, Y.N., Brown, J.A., Cohen, M.S., 2013. Non-negative matrix factorization of multimodal MRI, fMRI and phenotypic data reveals differential changes in default mode subnetworks in ADHD. NeuroImage.
Anderson, A., Han, D., Douglas, P.K., Bramen, J., Cohen, M.S., 2012. Real-time functional MRI classification of brain states using Markov-SVM hybrid models: peering inside the rt-fMRI black box. In: Machine Learning and Interpretation in Neuroimaging. Springer, pp. 242–255.
Anttila, P., Paatero, P., Tapper, U., Järvinen, O., 1995. Source identification of bulk wet deposition in F inland by positive matrix factorization. Atmosp. Environ. 29 (14), 1705–1718.
Bell, A.J., Sejnowski, T.J., 1995. An information-maximization approach to blind separation and blind deconvolution. Neural Comput. 7 (6), 1129–1159.
Bertsekas, D.P., 1999. Nonlinear Programming, second ed. Athena Scientific, Belmont, MA.
Bianciardi, M., Fukunaga, M., van Gelderen, P., de Zwart, J.A., Duyn, J.H., 2011. Negative BOLD-fMRI signals in large cerebral veins. J. Cerebral Blood Flow Metab. 31 (2), 401–412.
Breiman, L., 2001. Random forests. Machine Learn. 45 (1), 5–32.
Burges, C.J., 1998. A tutorial on support vector machines for pattern recognition. Data Mining Knowl. Discov. 2 (2), 121–167.
Calhoun, V., Pearlson, G., Adali, T., 2004. Independent component analysis applied to fMRI data: a generative model for validating results. J. VLSI Signal Process. Syst. Signal Image Video Technol. 37 (2–3), 281–291.
Calhoun, V.D., Potluru, V.K., Phlypo, R., Silva, R.F., Pearlmutter, B.A., Caprihan, A., Plis, S.M., Adalı, T., 2013. Independent component analysis for brain fMRI does indeed select for maximal independence. PLoS ONE 8 (8), e73309.
Cardoso, J.F., 1999. High-order contrasts for independent component analysis. Neural Comput. 11 (1), 157–192.
Cardoso, J.F., Souloumiac, A., 1993. Blind beamforming for non-Gaussian signals. In: IEEE Proceedings F (Radar and Signal Processing), vol. 140, IET, pp. 362–370.
Churchill, N.W., Spring, R., Afshin-Pour, B., Dong, F., Strother, S.C., 2015. An automated, adaptive framework for optimizing preprocessing pipelines in task-based functional MRI. PLOS ONE 10 (7), e0131520.
Churchill, N.W., Yourganov, G., Strother, S.C., 2014. Comparing within-subject classification and regularization methods in fMRI for large and small sample sizes. Hum. Brain Mapping 35 (9), 4499–4517.
Culbertson, C.S., Bramen, J., Cohen, M.S., London, E.D., Olmstead, R.E., Gan, J.J., Costello, M.R., Shulenberger, S., Mandelkern, M.A., Brody, A.L., 2011. Effect of bupropion treatment on brain activation induced by cigarette-related cues in smokers. Arch. Gen. Psychiatry 68 (5), 505–515.
Daubechies, I., Roussos, E., Takerkart, S., Benharrosh, M., Golden, C., D'ardenne, K., Richter, W., Cohen, J., Haxby, J., 2009. Independent component analysis for brain fMRI does not select for independence. Proc. Natl. Acad. Sci. U. S. A. 106 (26), 10415–10422.
Delorme, A., Makeig, S., 2004. EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics. J. Neurosci. Methods 134, 9–21.
Ding, X., Lee, J.H., Lee, S.W., 2012. Performance evaluation of nonnegative matrix factorization algorithms to estimate task-related neuronal activities from fMRI data. Magn. Reson. Imaging.
Douglas, P., Harris, S., Cohen, M., 2009. Naïve bayes classification of belief verses disbelief using event related neuroimaging data. NeuroImage 47, S80.
Douglas, P.K., Harris, S., Yuille, A., Cohen, M.S., 2011. Performance comparison of machine learning algorithms and number of independent components used in fMRI decoding of belief vs. disbelief. NeuroImage 56 (2), 544–553.
Douglas, P.K., Lau, E., Anderson, A., Head, A., Kerr, W., Wollner, M., Moyer, D., Li, W., Durnhofer, M., Bramen, J., et al., 2013. Single trial decoding of belief decision making from EEG and fMRI data using independent components features. Front. Hum. Neurosci., 7.
Eavani, H., Filipovych, R., Davatzikos, C., Satterthwaite, T.D., Gur, R.E., Gur, R.C., 2012. Sparse dictionary learning of resting state fMRI networks. In: International Workshop on IEEE Pattern Recognition in NeuroImaging (PRNI), pp. 73–76.
Ferdowsi, S., Abolghasemi, V., Makkiabadi, B., Sanei, S., 2011. A new spatially constrained NMF with application to fMRI. In: Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC, IEEE, pp. 5052–5055.
Ferdowsi, S., Abolghasemi, V., Sanei, S., 2010. A constrained NMF algorithm for BOLD detection in fMRI. In: IEEE International Workshop on Machine Learning for Signal Processing (MLSP), IEEE, pp. 77–82.
Friston, K.J., 1998. Modes or models: a critique on independent component analysis for fMRI. Trends Cogn. Sci. 2 (10), 373–375.
Griffanti, L., Salimi-Khorshidi, G., Beckmann, C.F., Auerbach, E.J., Douaud, G., Sexton, C.E., Zsoldos, E., Ebmeier, K.P., Filippini, N., Mackay, C.E., et al., 2014. ICA-based artefact removal and accelerated fMRI acquisition for improved resting state network imaging. NeuroImage 95, 232–247.
Guyon, I., Elisseeff, A., 2003. An introduction to variable and feature selection. J. Mach. Learn. Res. 3, 1157–1182.
Harel, N., Lee, S.P., Nagaoka, T., Kim, D.S., Kim, S.G., 2002. Origin of negative blood oxygenation level-dependent fMRI signals. J. Cereb. Blood Flow Metab. 22 (8), 908–917.
Haufe, S., Meinecke, F., Görgen, K., Dähne, S., Haynes, J.D., Blankertz, B., Bießmann, F., 2014. On the interpretation of weight vectors of linear models in multivariate neuroimaging. Neuroimage 87, 96–110.
Hyvärinen, A., Oja, E., 1997. A fast fixed-point algorithm for independent component analysis. Neural Comput. 9 (7), 1483–1492.

Hyvärinen, A., 1999. Fast and robust fixed-point algorithms for independent component analysis. IEEE Trans. Neural Netw. 10 (3), 626–634.
Jenkinson, M., Bannister, P., Brady, M., Smith, S., 2002. Improved optimization for the robust and accurate linear registration and motion correction of brain images. NeuroImage 17 (2), 825–841.
Kim, H., Park, H., 2007. Non-negative matrix factorization based on alternating non-negativity constrained least squares and active set method. Technical report, Technical Report GT-CSE-07-01. College of Computing, Georgia Institute of Technology.
Koldovsky, Z., Tichavsky, P., Oja, E., 2006. Efficient variant of algorithm fastica for independent component analysis attaining the cram&# 201; r-rao lower bound. IEEE Trans. Neural Netw. 17 (5), 1265–1277.
Lee, D.D., Seung, H.S., 1999. Learning the parts of objects by non-negative matrix factorization. Nature 401, 788–791.
Lee, D.D., Seung, H.S., 2001. Algorithms for non-negative matrix factorization. In: Advances in Neural Information Processing Systems, vol. 13, pp. 556–562.
Lee, H., Battle, A., Raina, R., Ng, A.Y., 2007. Efficient sparse coding algorithms. In: Advances in Neural Information Processing Systems, vol. 19, pp. 801–808.
Lee, K., Tak, S., Ye, J.C., 2011. A data-driven sparse GLM for fMRI analysis using sparse dictionary learning with MDL criterion. IEEE Trans. Med. Imaging 30 (5), 1076–1089.
Lee, K., Ye, J.C., 2010. Statistical parametric mapping of fMRI data using sparse dictionary learning. In: IEEE International Symposium on Biomedical Imaging: From Nano to Macro, IEEE, pp. 660–663.
Leonardi, N., Shirer, W.R., Greicius, M.D., Van De Ville, D., 2014. Disentangling dynamic networks: separated and joint expressions of functional connectivity patterns in time. Hum. Brain Mapp. 35 (12), 5984–5995.
Li, X.L., Adali, T., 2010. Independent component analysis by entropy bound minimization. IEEE Trans. Signal Process. 58 (10), 5151–5164.
Liaw, A., Wiener, M., 2002. Classification and regression by random forest. R News 2 (3), 18–22.
Lin, C.J., 2007. Projected gradient methods for non-negative matrix factorization. Neural Comput. 19, 2756–2779.
Liu, A., Chen, X., McKeown, M.J., Wang, Z.J., 2015. A sticky weighted regression model for time-varying resting-state brain connectivity estimation. IEEE Trans. Biomed. Eng. 62 (2), 501–510.
Logothetis, N.K., Sheinberg, D.L., 1996. Visual object recognition. Annu. Rev. Neurosci. 19 (1), 577–621.
Aharon, M., Elad, M., Bruckstein, A., 2006. The K-SVD: an algorithm for designing of overcomplete dictionaries for sparse representation. IEEE Trans. Signal Process. 54 (11), 4311–4322.
Mandelkow, H., de Zwart, J.A., Duyn, J.H., 2016. Linear discriminant analysis achieves high classification accuracy for the bold fMRI response to naturalistic movie stimuli. Front. Hum. Neurosci., 10.
McKeown, M.J., Hansen, L.K., Sejnowsk, T.J., 2003. Independent component analysis of functional MRI: what is signal and what is noise? Curr. Opin. Neurobiol. 13 (5), 620–629.
McKeown, M.J., Makeig, S., Brown, G.G., Jung, T.P., Kindermann, S.S., Bell, A.J., Sejnowski, T.J., 1997. Analysis of fMRI data by blind separation into independent spatial components, Tech. Rep., DTIC Document.
Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., Leisch, F., 2012. e1071: Misc Functions of the Department of Statistics (e1071), TU Wien. http://CRAN.R-project.org/package=e1071, r package version 1.6-1.
Moraschi, M., DiNuzzo, M., Giove, F., 2012. On the origin of sustained negative BOLD response. J. Neurophysiol. 108 (9), 2339–2342.
Berry, M.W., Browne, M., Langville, A.N., Paul Pauca, V., Plemmons, R.J., 2007. Algorithms and applications for approximate nonnegative matrix factorization. Comput. Stat. Data Anal. 52 (1), 155–173.
Naselaris, T., Kay, K.N., Nishimoto, S., Gallant, J.L., 2011. Encoding and decoding in FMRI. Neuroimage 56 (2), 400–410.
Olshausen, B.A., et al., 1996. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature 381 (6583), 607–609.
Paatero, P., Tapper, U., 1994. Positive matrix factorization: a non-negative factor model with optimal utilization of error estimates of data values. Environmetrics 5 (2), 111–126.
Palmer, S.E., 1977. Hierarchical structure in perceptual representation. Cogn. Psychol. 9 (4), 441–474.
Potluru, V.K., Calhoun, V.D., 2008. Group learning using contrast NMF: application to functional and structural MRI of schizophrenia. In: IEEE International Symposium on Circuits and Systems. ISCAS 2008, IEEE, pp. 1336–1339.
Rasmussen, P.M., Hansen, L.K., Madsen, K.H., Churchill, N.W., Strother, S.C., 2012. Model sparsity and brain pattern interpretation of classification models in neuroimaging. Pattern Recogn. 45 (6), 2085–2100.
Risk, B.B., Matteson, D.S., Ruppert, D., Eloyan, A., Caffo, B.S., 2013. An evaluation of independent component analyses with an application to resting-state fMRI. Biometrics.
Rubinstein, R., Bruckstein, A.M., Elad, M., 2010. Dictionaries for sparse representation modeling. Proc. IEEE 98 (6), 1045–1057.
Rubinstein, R., Elad, M., Zibulevsky, M., 2008. Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit. Technical Report – CS Technion.
Salimi-Khorshidi, G., Douaud, G., Beckmann, C.F., Glasser, M.F., Griffanti, L., Smith, S.M., 2014. Automatic denoising of functional MRI data: combining independent component analysis and hierarchical fusion of classifiers. NeuroImage 90, 449–468.
Sengupta, B., Stemmler, M., Laughlin, S.B., Niven, J.E., 2010. Action potential energy efficiency varies among neuron types in vertebrates and invertebrates. PLoS Comput. Biol. 6 (7), e1000840.
Smith, A.T., Williams, A.L., Singh, K.D., 2004. Negative BOLD in the visual cortex: evidence against blood stealing. Hum. Brain Mapp. 21 (4), 213–220.
Smith, S.M., 2002. Fast robust automated brain extraction. Hum. Brain Mapp. 17 (3), 143–155.
Smith, S.M., Fox, P.T., Miller, K.L., Glahn, D.C., Fox, P.M., Mackay, C.E., Filippini, N., Watkins, K.E., Toro, R., Laird, A.R., et al., 2009. Correspondence of the brain's functional architecture during activation and rest. Proc. Natl. Acad. Sci. U. S. A. 106 (31), 13040–13045.
Spratling, M.W., 2014. Classification using sparse representations: a biologically plausible approach. Biol. Cybern. 108 (1), 61–73.
the ICA and BSS group, U.o.H., 2014. The fastica package for matlab. http://research.ics.aalto.fi/ica/fastica/.
Varoquaux, G., Raamana, P.R., Engemann, D.A., Hoyos-Idrobo, A., Schwartz, Y., Thirion, B., 2016. Assessing and tuning brain decoders: cross-validation, caveats, and guidelines. NeuroImage.
Wachsmuth, E., Oram, M., Perrett, D., 1994. Recognition of objects and their component parts: responses of single units in the temporal cortex of the macaque. Cereb. Cortex 4 (5), 509–522.
Wang, X., Tian, J., Li, X., Dai, J., Ai, L., 2004. Detecting brain activations by constrained non-negative matrix factorization from task-related BOLD fMRI. In: Medical Imaging 2004, International Society for Optics and Photonics, pp. 675–682.