Learning brain regions via large-scale online structured sparse dictionary-learning

Elvis Dohmatob, Arthur Mensch, Gaël Varoquaux, Bertrand Thirion

Learning brain regions via large-scale online structured sparse dictionary-learning

Elvis Dohmatob, Arthur Mensch, Gael Varoquaux, Bertrand Thirion
firstname.lastname@inria.fr
Parietal Team, INRIA / CEA, Université Paris-Saclay, France

Abstract

We propose a multivariate online dictionary-learning method for obtaining decompositions of brain images with structured and sparse components (aka atoms). Sparsity is to be understood in the usual sense: the dictionary atoms are constrained to contain mostly zeros. This is imposed via an L1-norm constraint. By "structured", we mean that the atoms are piece-wise smooth and compact, thus making up blobs, as opposed to scattered patterns of activation. We propose to use a Sobolev (Laplacian) penalty to impose this type of structure. Combining the two penalties, we obtain decompositions that properly delineate brain structures from functional images. This non-trivially extends the online dictionary-learning work of Mairal et al. (2010), at the price of only a factor of 2 or 3 on the overall running time. Just like the Mairal et al. (2010) reference method, the online nature of our proposed algorithm allows it to scale to arbitrarily sized datasets. Experiments on brain data show that our proposed method extracts structured and denoised dictionaries that are more intepretable and better capture inter-subject variability in small medium, and large-scale regimes alike, compared to state-of-the-art models.

1 Introduction

In neuro-imaging, inter-subject variability is often handled as a statistical residual and discarded. Yet there is evidence that it displays structure and contains important information. Univariate models are ineffective both computationally and statistically due to the large number of voxels compared to the number of subjects. Likewise, statistical analysis of weak effects on medical images often relies on defining regions of interests (ROIs). For instance, pharmacology with Positron Emission Tomography (PET) often studies metabolic processes in specific organ sub-parts that are defined from anatomy. Population-level tests of tissue properties, such as diffusion, or simply their density, are performed on ROIs adapted to the spatial impact of the pathology of interest. In functional brain imaging, e.g function magnetic resonance imaging (fMRI), ROIs must be adapted to the cognitive process under study, and are often defined by the very activation elicited by a closely related process [20]. ROIs can boost statistical power by reducing multiple comparisons that plague image-based statistical testing. If they are defined to match spatially the differences to detect, they can also improve the signal-to-noise ratio by averaging related signals. However, the crux of the problem is how to define these ROIs in a principled way. Indeed, standard approaches to region definition imply a segmentation step. Segmenting structures on first-level statistical maps, as in fMRI, typically yields meaningful units, but is limited by the noise inherent to these maps. Relying on a different imaging modality hits cross-modality correspondence problems.

Sketch of our contributions. In this manuscript, we propose to use the *variability* of the statistical maps across the population to define regions. This idea is reminiscent of clustering approaches, that have been employed to define spatial units for quantitative analysis of information as diverse as brain fiber tracking, brain activity, brain structure, or even imaging-genetics. See [22, 16] and references therein. The key idea is to group together features –voxels of an image, vertices on a mesh, fiber tracts–

29th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

based on the quantity of interest, to create regions –or fiber bundles– for statistical analysis. However, unlike clustering that models each observation as an instance of a cluster, we use a model closer to the signal, where each observation is a linear mixture of several signals. The model is closer to mode finding, as in a principal component analysis (PCA), or an independent component analysis (ICA), often used in brain imaging to extract functional units [5]. Yet, an important constraint is that the modes should be sparse and spatially-localized. For this purpose, the problem can be reformulated in a dictionary-learning, as ICA/PCA, with appropriate spatial and sparse penalties [26, 1].

We propose a multivariate online dictionary-learning method for obtaining decompositions with structured and sparse components (aka atoms). Sparsity is to be understood in the usual sense: the atoms contain mostly zeros. This is imposed via an L1 penalty on the atoms. By "structured", we mean that the atoms are piece-wise smooth and compact, thus making up blobs, as opposed to scattered patterns of activation. We impose this type of structure via a Laplacian penalty on the dictionary atoms. Combining the two penalties, we therefore obtain decompositions that are closer to known functional organization of the brain. This non-trivially extends the online dictionary-learning / dictionary-learning work [18], with only a factor of 2 or 3 on the running time. By means of experiments on a large public dataset, we show the improvements brought by the spatial regularization with respect to traditional $L1$-regularized dictionary learning. We also provide a concise study of the impact of hyper-parameter selection on this problem and describe the optimality regime, based on relevant criteria (reproducibility, captured variability, explanatory power in prediction problems).

2 Smooth Sparse Online Dictionary-Learning (Smooth-SODL)

Consider a stack $\mathbf{X} \in \mathbb{R}^{n \times p}$ of n subject-level brain images $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n$ each of shape $n_1 \times n_2 \times n_3$, seen as p-dimensional row vectors –with $p = n_1 \times n_2 \times n_3$, the number of voxels. These could be images of fMRI activity patterns like statistical parametric maps of brain activation, raw pre-registered (into a common coordinate space) fMRI time-series, PET images, etc. We would like to decompose these images as a product of $k \leq \min(n, p)$ component maps (aka latent factors or dictionary atoms) $\mathbf{V}^1, \ldots, \mathbf{V}^k \in \mathbb{R}^{p \times 1}$ and modulation coefficients $\mathbf{U}_1, \ldots, \mathbf{U}_n \in \mathbb{R}^{k \times 1}$ called *codes* (one k-dimensional code per sample point), i.e

$$\mathbf{X}_i \approx \mathbf{V}\mathbf{U}_i, \text{ for } i = 1, 2, \ldots, n \tag{1}$$

where $\mathbf{V} := [\mathbf{V}^1 | \ldots | \mathbf{V}^k] \in \mathbb{R}^{p \times k}$, an unknown dictionary to be estimated. Typically, $p \sim 10^5$ – 10^6 (in full-brain high-resolution fMRI) and $n \sim 10^2$ – 10^5 (for example, in considering all the 500 subjects and all the about 15 –20 functional tasks of the Human Connectome Project dataset [11]). Our work handles the extreme case where both n and p are large (massive-data setting). It is reasonable then to only consider under-complete dictionaries: $k \leq \min(n, p)$. Typically, we use $k \sim 50$ or 100 components. It should be noted that online optimization is not only crucial in the case where n/p is big; it is relevant whenever n is large, leading to prohibitive memory issues irrespective of how big or small p is.

As explained in the introduction, we would want the component maps (aka dictionary atoms) $\mathbf{V}^j$ to be sparse and spatially smooth. A principled way to achieve such a goal is to impose a boundedness constraint on ℓ_1-like norms of these maps to achieve sparsity and simultaneously impose smoothness by penalizing their Laplacian. Thus, we propose the following penalized dictionary-learning model

$$\min_{\mathbf{V} \in \mathbb{R}^{p \times k}} \left(\lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} \min_{\mathbf{U}_i \in \mathbb{R}^k} \frac{1}{2} \|\mathbf{X}_i - \mathbf{V}\mathbf{U}_i\|_2^2 + \frac{1}{2} \alpha \|\mathbf{U}_i\|_2^2 \right) + \gamma \sum_{j=1}^{k} \Omega_{\text{Lap}}(\mathbf{V}^j). \tag{2}$$

subject to $\mathbf{V}^1, \ldots, \mathbf{V}^k \in \mathcal{C}$

The ingredients in the model can be broken down as follows:

- Each of the terms $\max_{\mathbf{U}_i \in \mathbb{R}^k} \frac{1}{2} \|\mathbf{X}_i - \mathbf{V}\mathbf{U}_i\|_2^2$ measures how well the current dictionary $\mathbf{V}$ explains data $\mathbf{X}_i$ from subject i. The Ridge penalty term $\phi(\mathbf{U}_i) \equiv \frac{1}{2} \alpha \|\mathbf{U}_i\|_2^2$ on the codes amounts to assuming that the energy of the decomposition is spread across the different samples. In the context of a specific neuro-imaging problem, if there are good grounds to assume that each sample / subject should be sparsely encoded across only a few atoms of the dictionary, then we can use the L1 penalty $\phi(\mathbf{U}_i) := \alpha \|\mathbf{U}_i\|_1$ as in [18]. We note that in

contrast to the L1 penalty, the Ridge leads to stable codes. The parameter $\alpha > 0$ controls the amount of penalization on the codes.

- The constraint set $\mathcal{C}$ is a sparsity-inducing compact simple[1] convex subset of $\mathbb{R}^p$ like an L1-ball $\mathbb{B}_{p,\ell_1}(\tau)$ or a simplex $\mathcal{S}_p(\tau)$, defined respectively as

$$\mathbb{B}_{p,\ell_1}(\tau) := \{\mathbf{v} \in \mathbb{R}^p \text{ s.t } |\mathbf{v}_1| + |\mathbf{v}_2| + \ldots + |\mathbf{v}_p| \leq \tau\},$$

and $\mathcal{S}_p(\tau) := \mathbb{B}_{p,\ell_1}(\tau) \cap \mathbb{R}^p_+$. Other choices (e.g ElasticNet ball) are of course possible. The radius parameter $\tau > 0$ controls the amount of sparsity: smaller values lead to sparser atoms.

- Finally, Ω_{Lap} is the 3D Laplacian regularization functional defined by

$$\Omega_{\text{Lap}}(\mathbf{v}) := \frac{1}{2}\sum_{k=1}^{p}(\nabla_x \mathbf{v})_k^2 + (\nabla_y \mathbf{v})_k^2 + (\nabla_z \mathbf{v})_k^2 = \frac{1}{2}\mathbf{v}^T \Delta \mathbf{v} \geq 0, \ \forall \mathbf{v} \in \mathbb{R}^p, \tag{3}$$

∇_x being the discrete spatial gradient operator along the x-axis (a p-by-p matrix), ∇_y along the y-axis, etc., and $\Delta := \nabla^T \nabla$ is the p-by-p matrix representing the discrete Laplacian operator. This penalty is meant to impose blobs. The regularization parameter $\gamma \geq 0$ controls how much regularization we impose on the atoms, compared to the reconstruction error.

The above formulation, which we dub *Smooth Sparse Online Dictionary-Learning* (Smooth-SODL) is inspired by, and generalizes the standard dictionary-learning / dictionary-learning framework of [18] –henceforth referred to as *Sparse Online Dictionary-Learning* (SODL); setting $\gamma = 0$, we recover SODL [18].

3 Estimating the model

3.1 Algorithms

The objective function in problem (2) is separately convex and block-separable w.r.t each of $\mathbf{U}$ and $\mathbf{V}$ but is not jointly convex in $(\mathbf{U}, \mathbf{V})$. Also, it is continuously differentiable on the constraint set, which is compact and convex. Thus by classical results (e.g Bertsekas [6]), the problem can be solved via Block-Coordinate Descent (BCD) [18]. Reasoning along the lines of [17], we derive that the BCD iterates are as given in Alg. 1. A crucial advantage of using a BCD scheme is that it is parameter free: there is not step size to tune. The resulting algorithm Alg. 1, is adapted from [18]. It relies on Alg. 2 for performing the structured dictionary updates, the details of which are discussed below.

Algorithm 1 Online algorithm for the dictionary-learning problem (2)

Require: Regularization parameters $\alpha, \gamma > 0$; initial dictionary $\mathbf{V} \in \mathbb{R}^{p \times k}$, number of passes / iterations T on the data.

1: $\mathbf{A}_0 \leftarrow 0 \in \mathbb{R}^{k \times k}$, $\mathbf{B}_0 \leftarrow 0 \in \mathbb{R}^{p \times k}$ (historical "sufficient statistics")
2: **for** $t = 1$ to T **do**
3: Empirically draw a sample point $\mathbf{X}_t$ at random.
4: Code update: Ridge-regression (via SVD of current dictionary $\mathbf{V}$)

$$\mathbf{U}_t \leftarrow \text{argmin}_{\mathbf{u} \in \mathbb{R}^k} \frac{1}{2}\|\mathbf{X}_t - \mathbf{V}\mathbf{u}\|_2^2 + \frac{1}{2}\alpha\|\mathbf{u}\|_2^2. \tag{4}$$

5: Rank-1 updates: $\mathbf{A}_t \leftarrow \mathbf{A}_{t-1} + \mathbf{U}_t\mathbf{U}_t^T$, $\mathbf{B}_t \leftarrow \mathbf{B}_{t-1} + \mathbf{X}_t\mathbf{U}_t^T$
6: BCD dictionary update: Compute update for dictionary $\mathbf{V}$ using Alg. 2.
7: **end for**

Update of the codes: Ridge-coding. The Ridge sub-problem for updating the codes

$$\mathbf{U}_t = (\mathbf{V}^T\mathbf{V} + \alpha\mathbf{I})^{-1}\mathbf{V}^T\mathbf{X}_t \tag{5}$$

is computed via an SVD of the current dictionary $\mathbf{V}$. For $\alpha \approx 0$, $\mathbf{U}_t$ reduces to the orthogonal projection of $\mathbf{X}_t$ onto the image of the current dictionary $\mathbf{V}$. As in [18], we speed up the overall algorithm by sampling mini-batches of η samples $\mathbf{X}_t, \ldots, X_\eta$ and compute the corresponding codes $\mathbf{U}_1$, $\mathbf{U}_2$, ..., $\mathbf{U}_\eta$ at once. We typically use we use mini-batches of size $\eta \sim 20$.

[1] Mainly in the sense that the Euclidean projection onto $\mathcal{C}$ should be easy to compute.

BCD dictionary update for the dictionary atoms. Let us define time-varying matrices $\mathbf{A}_t := \sum_{i=1}^{t} \mathbf{U}_i \mathbf{U}_i^T \in \mathbb{R}^{k \times k}$ and $\mathbf{B}_t := \sum_{i=1}^{t} \mathbf{X}_i \mathbf{U}_i^T \in \mathbb{R}^{p \times k}$, where $t = 1, 2, \ldots$ denotes time. We fix the matrix of codes $\mathbf{U}$, and for each j, consider the update of the jth dictionary atom, with all the other atoms $\mathbf{V}^{k \neq j}$ kept fixed. The update for the atom $\mathbf{V}^j$ can then be written as

$$\begin{aligned}
\mathbf{V}^j &= \text{argmin}_{\mathbf{v} \in \mathcal{C}, \mathbf{V}=[\mathbf{V}^1|\ldots|\mathbf{v}|\ldots|\mathbf{V}^k]} \frac{1}{t} \sum_{i=1}^{t} \left(\frac{1}{2} \|\mathbf{X}_i - \mathbf{V}\mathbf{U}_i\|_2^2 \right) + \gamma \Omega_{\text{Lap}}(\mathbf{v}) \\
&= \text{argmin}_{\mathbf{v} \in \mathcal{C}, \mathbf{V}=[\mathbf{V}^1|\ldots|\mathbf{v}|\ldots|\mathbf{V}^k]} \left(\sum_{i=1}^{t} \frac{1}{2} \|\mathbf{X}_i - \mathbf{V}\mathbf{U}_i\|_2^2 \right) + \gamma t \Omega_{\text{Lap}}(\mathbf{v}) \qquad (6) \\
&= \text{argmin}_{\mathbf{v} \in \mathcal{C}} F_{\gamma(\mathbf{A}_t[j,j]/t)^{-1}}(\mathbf{v}, \underbrace{\mathbf{V}^j + \mathbf{A}_t[j,j]^{-1}(\mathbf{B}_t^j - \mathbf{V}\mathbf{A}_t^j)}_{\text{refer to [18] for the details}}),
\end{aligned}$$

where $F_{\tilde{\gamma}}(\mathbf{v}, \mathbf{a}) \equiv \frac{1}{2}\|\mathbf{v} - \mathbf{a}\|_2^2 + \tilde{\gamma}\Omega_{\text{Lap}}(\mathbf{v}) = \frac{1}{2}\|\mathbf{v} - \mathbf{a}\|_2^2 + \frac{1}{2}\tilde{\gamma}\mathbf{v}^T \Delta \mathbf{v}$.

Algorithm 2 BCD dictionary update with Laplacian prior

Require: $\mathbf{V} = [\mathbf{V}^1|\ldots|\mathbf{V}^k] \in \mathbb{R}^{p \times k}$ (input dictionary),
$\mathbf{A} = [\mathbf{A}^1|\ldots|\mathbf{A}^k] \in \mathbb{R}^{k \times k}$, $\mathbf{B}_t = [\mathbf{B}_t^1|\ldots|\mathbf{B}_t^k] \in \mathbb{R}^{p \times k}$ (history)
while stopping criteria not met, **do**
  for $j = 1$ to r **do**
    Fix the code $\mathbf{U}$ and all atoms $k \neq j$ of the dictionary $\mathbf{V}$ and then update $\mathbf{V}^j$ as follows

$$\mathbf{V}^j \leftarrow \text{argmin}_{\mathbf{v} \in \mathcal{C}} F_{\gamma(\mathbf{A}_t[j,j]/t)^{-1}}(\mathbf{v}, \mathbf{V}^j + \mathbf{A}_t[j,j]^{-1}(\mathbf{B}_t^j - \mathbf{V}\mathbf{A}_j)) \qquad (7)$$

    (See below for details on the derivation and the resolution of this problem)
  end for
end while

Problem (6) is thus a compactly-constrained minimization of the 1-strongly-convex quadratic functions $F_{\tilde{\gamma}}(., \mathbf{a}) : \mathbb{R}^p \to \mathbb{R}$ defined above. This problem can further be identified with a denoising instance (i.e in which the design matrix or deconvolution operator is the identity operator) of the GraphNet model [13, 15]. Fast first-order methods like FISTA[4] with optimal rates $\mathcal{O}(L/\sqrt{\epsilon})$ are available[2] for solving such problems to arbitrary precision $\epsilon > 0$. One computes the Lipschitz constant to be $L_{F_{\tilde{\gamma}}(.,\mathbf{a})} \equiv 1 + \tilde{\gamma} L_{\Omega_{\text{Lap}}} = 1 + 4D\tilde{\gamma}$, where as before, D is the number of spatial dimensions with $D = 3$ for volumic images. One should also mention that under certain circumstances, it is possible to perform the dictionary updates in the Fourier domain, via FFT. This alternative approach is developed in Appendix B.

Finally, one notes that, since constraints in problem (2) are separable in the dictionary atoms $\mathbf{V}^j$, the BCD dictionary-update algorithm Alg. 2 is guaranteed to converge to a global optimum, at each iteration [6, 18].

How difficult is the dictionary update for our proposed model ? A favorable property of the vanilla dictionary-learning [18] is that the BCD dictionary updates amount to Euclidean projections onto the constraint set $\mathcal{C}$, which can be easily computed for a variety of choices (simplexes, closed convex balls, etc.). One may then ask: do we retain a comparable algorithmic simplicity even with the additional Laplacian terms $\Omega_{\text{Lap}}(\mathbf{V}^j)$? The short answer is yes: empirically, we found that 1 or 2 iterations of FISTA[4] are sufficient reach an accuracy of 10^{-6} in problem (6), which is sufficient to obtain a good decomposition in the overall algorithm.

However, choosing γ "too large" will provably cause the dictionary updates to eventually take forever to run. Indeed, the Lipschitz constant in problem (6) is $L_t = 1 + 4D\gamma(\mathbf{A}_t[j,j]/t)^{-1}$, which will blow-up (leading to arbitrarily small step-sizes) unless γ is chosen so that

$$\gamma = \gamma_t = \mathcal{O}\left(\max_{1 \leq j \leq k} \mathbf{A}_t[j,j] \right) = \mathcal{O}\left(\max_{1 \leq j \leq k} \sum_{i=1}^{t} \|\mathbf{U}^j\|_2^2 / t \right) = \mathcal{O}(\|A_t\|_{\infty,\infty}/t). \qquad (8)$$

[2]For example, see [9, 25], implemented as part of the *Nilearn* open-source library Python library [2].

Finally, the Euclidean projections onto the L1 ball $\mathcal{C}$ can be computed exactly in linear-time $\mathcal{O}(p)$ (see for example [7, 10]). The dictionary atoms j are repeatedly cycled and problem (6) solved. All in all, in practice we observe that a single iteration is sufficient for the dictionary update sub-routine in Alg. 2 to converge to a qualitatively good dictionary.

Convergence of the overall algorithm. The Convergence of our algorithm (to a local optimum) is guaranteed since all hypotheses of [18] are satisfied. For example, assumption **(A)** is satisfied because fMRI data are naturally compactly supported. Assumption **(C)** is satisfied since the ridge-regression problem (4) has a unique solution. More details are provided in Appendix A.

3.2 Practical considerations

Hyper-parameter tuning. Parameter-selection in dictionary-learning is known to be a difficult unsolved problem [18, 17], and our proposed model (2) is not an exception to this rule. We did an extensive study of the quality of estimated dictionary varies with the model hyper-parameters (α, γ, τ). The data experimental setup is described in Section 5. The results are presented in Fig. 1. We make the following observations: Taking the sparsity parameter τ in (2) too large leads to dense atoms that perfectly explain the data but are not very intepretable. Taking it too small leads to overly sparse maps that barely explain the data. This normalized sparsity metric (small is better, *ceteris paribus*) is defined as the mean ratio $\|\mathbf{V}^j\|_1/\|\mathbf{V}^j\|_2$ over the dictionary atoms.

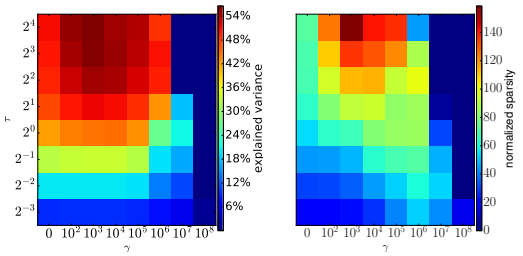


Figure 1: **Influence of model parameters.** In the experiments, α was chosen according to (9). **Left:** Percentage explained variance of the decomposition, measured on left-out data split. **Right:** Average normalized sparsity of the dictionary atoms.

Concerning the α parameter, inspired by [27], we have found the following time-varying data-adaptive choice for the α parameter to work very well in practice:

$$\alpha = \alpha_t \sim t^{-1/2}. \tag{9}$$

Likewise, care must be taken in selecting the Laplacian regularization parameter γ. Indeed taking it too small amounts to doing vanilla dictionary-learning model [18]. Taking it too large can lead to degenerate maps, as the spatial regularization then dominates the reconstruction error (data fidelity) term. We find that there is a safe range of the parameter pair (γ, τ) in which a good compromise between the sparsity of the dictionary (thus its intepretability) and its explanation power of the data can be reached. See Fig. 1. K-fold cross-validation with explained variance metric was retained as a good strategy for setting the Laplacian regularization γ parameter and the sparsity parameter τ.

Initialization of the dictionary. Problem (2) is non-convex jointly in $(\mathbf{U}, \mathbf{V})$, and so initialization might be a might be a crucial issue. However, in our experiments, we have observed that even randomly initialized dictionaries eventually produce sensible results that do not jitter much across different runs of the same experiment.

4 Related works

While there exist algorithms for online sparse dictionary-learning that are very efficient in large-scale settings (for example [18], or more recently [19]) imposing spatial structure introduces couplings in the corresponding optimization problem [9]. So far, spatially-structured decompositions have been solved by very slow alternated optimization [26, 1]. Notably, structured priors such as TV-L1 [3] minimization, were used by [1] to extract data-driven state-of-the-art atlases of brain function. However, alternated minimization is very slow, and large-scale medical imaging has shifted to online solvers for dictionary-learning like [18] and [19]. These do not readily integrate structured penalties. As a result, the use of structured decompositions has been limited so far, mostly due to the computational cost of the ensuing algorithms. Our approach instead uses a Laplacian penalty

to impose spatial structure at a very minor cost and adapts the online-learning dictionary-learning framework [18], resulting in a fast and scalable structured decomposition. Second, the approach in [1] though very novel, is heuristic, as it does not come with theoretical guarantees. In contrast, our method enjoys the same convergence guarantees and comparable numerical complexity as the basic unstructured online dictionary-learning [18].

Finally, one should also mention [24] which introduced an online group-level functional brain mapping strategy for differentiating regions reflecting the variety of brain network configurations observed a the population, by learning a sparse-representation of these in the spirit of [18].

5 Experiments

Setup. Our experiments were done on task fMRI data from 500 subjects from the HCP –Human Connectome Project– dataset [11]. These task fMRI data were acquired in an attempt to assess major domains that are thought to sample the diversity of neural systems of interest in functional connectomics. We studied the activation maps related to a task that involves language (story understanding) and mathematics (mental computation). This particular task is expected to outline number, attentional and language networks, but the variability modes observed in the population cover even wider systems. For the experiments, mass-univariate General Linear Models (GLMs) [12] for $n = 500$ subjects were estimated for the *Math vs Story* contrast (language protocol), and the corresponding full-brain Z-score maps each containing $p = 2.6 \times 10^5$ voxels, were used as the input data $\mathbf{X} \in \mathbb{R}^{n \times p}$, and we sought a decomposition into a dictionary of $k = 40$ atoms (components). The input data $\mathbf{X}$ were shuffled and then split into two groups of the same size.

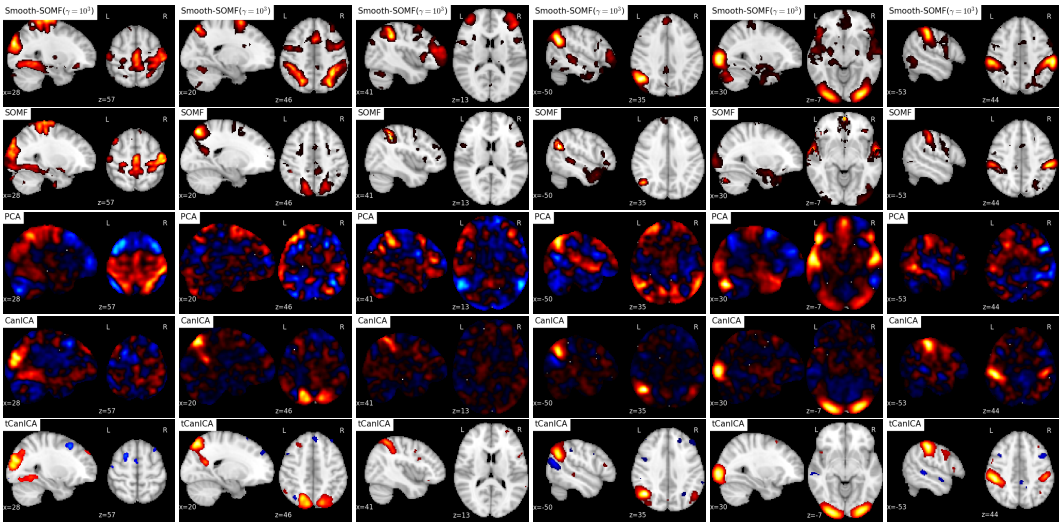
Models compared and metrics. We compared our proposed Smooth-SODL model (2) against both the Canonical ICA –CanICA [23], a single-batch multi-subject PCA/ICA-based method, and the standard SODL (sparse online dictionary-learning) [18]. While the CanICA model accounts for subject-to-subject differences, one of its major limitations is that it does not model spatial variability across subjects. Thus we estimated the CanICA components on smoothed data: isotropic FWHM of 6mm, a necessary preprocessing step for such methods. In contrast, we did no pre-smoothing for the SODL of Smooth-SODL models. The different models were compared across a variety of qualitative and quantitative metrics: visual quality of the dictionaries obtained, explained variance, stability of the dictionary atoms, their reproducibility, performance of the dictionaries in predicting behavioral scores (IQ, picture vocabulary, reading proficiency, etc.) shipped with the HCP data [11]. For both SODL[18] and our proposed Smooth-SODL model, the constraint set for the dictionary atoms was taken to be a simplex $\mathcal{C} := \mathcal{S}_p(\tau)$ (see section 2 for definition). The results of these experiments are presented in Fig. 2 and Tab. 1.

6 Results

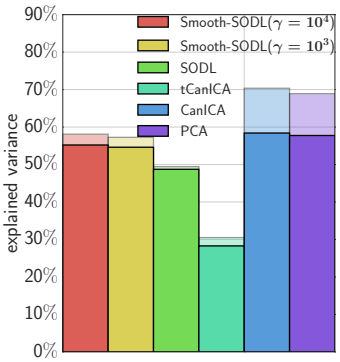
Running time. On the computational side, The standard dictionary-learning SODL algorithm [18] with a batch size of $\eta = 20$ took about 110s (≈ 1.7 minutes) to run, whilst with the same batch size, our proposed Smooth-SODL model (2) implemented in Alg. 1 took 340s (≈ 5.6 minutes), which is slightly less than $3\times$ slower than SODL. Finally, CanICA [23] for this experiment took 530s (≈ 8.8 minutes) to run, which is about $5\times$ slower than the SODL model and $1.6\times$ slower than our proposed Smooth-SODL (2) model. All experiments were run on a single CPU of a modern laptop.

Qualitative assessment of dictionaries. As can be seen in Fig. 2*(a)*, all methods recover dictionary atoms that represent known functional brain organization; notably the dictionaries all contain the well-known executive control and attention networks, at least in part. Vanilla dictionary-learning leverages the denoising properties of the L1 sparsity constraint, but the voxel clusters are not very structured. For, example most blobs are surrounded with a thick ring of very small nonzero values. In contrast, our proposed regularization model leverages both sparse and structured dictionary atoms, which are more spatially structured and less noisy.

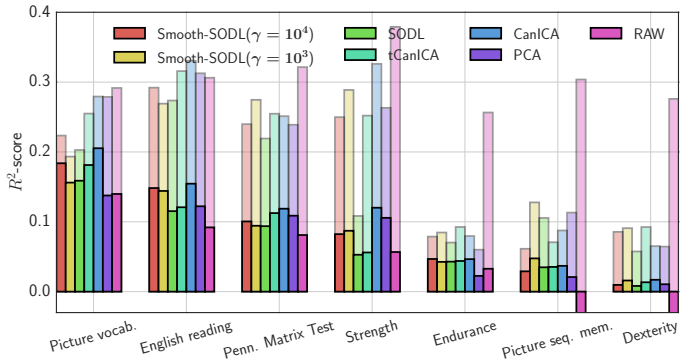
In contrast to both SODL and Smooth-SODL, CanICA [23] is an ICA-based method which enforces no notion of sparsity whatsoever. The result are therefore dense and noisy dictionary atoms that explain the data very well (Fig. 2*(b)* but which are completely unintepretable. In a futile attempt to remedy the situation, in practice such PCA/ICA-based methods (including FSL's MELODIC tool



(a) **Qualitative comparison of the estimated dictionaries.** Each column represents an atom of the estimated dictionary, where atoms from the different models (the rows of the plots) have been matched via a Hungarian algorithm. Here, we only show a limited number of the most "intepretable" atoms. Notice how the major structures in each atom are reproducible across the different models. Maps corresponding to hard-thresholded CanICA [23] components have also been included, and have been called tCanICA. In contrast, the maps from the SODL [18] and our proposed Smooth-SODL (2) were not been thresholded.



(b) **Mean explained variance** of the different models on both training data and test (left-out) data. **N.B.:** Bold bars represent performance on **test** set while faint bars in the background represent performance on **train** set.



(c) **Predicting behavioral variables** of the HCP[11] dataset using subject-level Z-maps. **N.B.:** Bold bars represent performance on **test** set while faint bars in the background represent performance on **train** set.

Figure 2: **Main results.** Benchmarking our proposed Smooth-SODL (2) model against competing state-of-the-art methods like SODL (sparse online dictionary-learning) [18] and CanICA [23].

[21]) are hard-thresholded in order to see information. For CanICA, the hard-thresholded version has been named tCanICA in Fig. 2. That notwithstanding, notice how the major structures (parietal lobes, sulci, etc.) in each atom are reproducible across the different models.

Stability-fidelity trade-offs. PCA/ICA-based methods like CanICA [23] and MELODIC [21] are the optimal linear decomposition method to maximize explained variance on a dataset. On the training set, CanICA [23] outperforms all others algorithms with about 66% (resp. 50% for SODL[18] and 58% for Smooth-SODL) of explained variance on the training set, and 60% (resp. 49% for SODL and **55%** for Smooth-SODL) on left-out (test) data. See Fig. 2*(b)*. However, as noted in the above paragraph, such methods lead to dictionaries that are hardly intepretable and thus the user must recourse to some kind of post-processing hard-thresholding step, which destroys the estimated model. More so, assessing the stability of the dictionaries, measured by mean correlation between corresponding

atoms, across different splits of the data, CanICA [23] scores a meager 0.1, whilst the hard-thresholded version tCanICA obtains 0.2, compared to **0.4** for Smooth-SODL and 0.1 for SODL.

Is spatial regularization really needed ? As rightly pointed out by one of the reviewers, one does not need spatial regularization if data are abundant (like in the HCP). So we computed learning curves of mean explained variance (EV) on test data, as a function of the amount training data seen by both Smooth-SODL and SODL [18] (Table 1). In the beginning of the curve, our proposed spatially regularized Smooth-SODL model starts off with more than 31% explained variance (computed on 241 subjects), after having pooled only 17 subjects. In contrast, the vanilla SODL model [18] scores a meager 2% explained variance; this corresponds to a 14-fold gain of Smooth-SODL over SODL. As more and more that are pooled, both models explain more variance, and the gap between Smooth-SODL and SODL reduces, and both models perform comparably asymptotically.

Nb. subjects pooled	mean EV for vanilla SODL [18]	Smooth-SODL (2)	gain factor
17	**2%**	**31%**	**13.8**
92	37%	50%	1.35
167	47%	54%	1.15
241	49%	**55%**	1.11

Table 1: **Learning-curve** for boost in explained variance of our proposed Smooth-SODL model over the reference SODL model. Note the reduction in the explained variance gain as more data are pooled.

Thus our proposed Smooth-SODL method extracts structured denoised dictionaries that better capture inter-subject variability in small, medium, and large-scale regimes alike.

Prediction of behavioral variables. If Smooth-SODL captures the patterns of inter-subject variability, then it should be possible to predict cognitive scores $\mathbf{y}$ like picture vocabulary, reading proficiency, math aptitude, etc. (the behavioral variables are explained in the HCP wiki [14]) by projecting new subjects' data into this learned low-dimensional space (via solving the ridge problem (4) for each sample $\mathbf{X}_t$), without loss of performance compared with using the raw Z-values values $\mathbf{X}$. Let RAW refer to the direct prediction of targets $\mathbf{y}$ from $\mathbf{X}$, using the top 2000 most voxels most correlated with the target variable. Results of for the comparison are shown in Fig. 2*(c)*. Only variables predicted with a a positive mean (across the different methods and across subjects) R-score are reported. We see that the RAW model, as expected over-fits drastically, scoring an R^2 of 0.3 on training data and only 0.14 on test data. Overall, for this metric CanICA performs best than all the other models in predicting the different behavioral variables on test data. However, our proposed Smooth-SODL model outperforms both SODL [18] and tCanICA, the thresholded version of CanICA.

7 Concluding remarks

To extract structured functionally discriminating patterns from massive brain data (i.e data-driven atlases), we have extended the online dictionary-learning framework first developed in [18], to learn structured regions representative of brain organization. To this end, we have successfully augmented [18] with a Laplacian prior on the component maps, while conserving the low numerical complexity of the latter. Through experiments, we have shown that the resultant model –Smooth-SODL model (2)– extracts structured and denoised dictionaries that are more intepretable and better capture inter-subject variability in small medium, and large-scale regimes alike, compared to state-of-the-art models. We believe such online multivariate online methods shall become the de facto way do dimensionality reduction and ROI extraction in future.

Implementation. The authors' implementation of the proposed SS0MF (2) model will soon be made available as part of the *Nilearn* package [2].

Acknowledgment. This work has been funded by EU FP7/2007-2013 under grant agreement no. 604102, Human Brain Project (HBP) and the iConnectome Digiteo. We would also like to thank the Human Connectome Projection for making their wonderful data publicly available.

References

[1] A. Abraham et al. "Extracting brain regions from rest fMRI with Total-Variation constrained dictionary learning". In: *MICCAI*. 2013.

[2] A. Abraham et al. "Machine learning for neuroimaging with scikit-learn". In: *Frontiers in Neuroinformatics* (2014).

[3] L. Baldassarre, J. Mourao-Miranda, and M. Pontil. "Structured sparsity models for brain decoding from fMRI data". In: *PRNI*. 2012.

[4] A. Beck and M. Teboulle. "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems". In: *SIAM J. Imaging Sci.* 2 (2009).

[5] C. F. Beckmann and S. M. Smith. "Probabilistic independent component analysis for functional magnetic resonance imaging". In: *Trans Med. Im.* 23 (2004).

[6] D. P. Bertsekas. *Nonlinear programming*. Athena Scientific, 1999.

[7] L. Condat. "Fast projection onto the simplex and the ℓ_1 ball". In: *Math. Program.* (2014).

[8] Y. Dai. "Fast Algorithms for Projection on an Ellipsoid". In: *SIAM J. Optim.* 16 (2006).

[9] E. Dohmatob et al. "Benchmarking solvers for TV-l1 least-squares and logistic regression in brain imaging". In: *PRNI*. IEEE. 2014.

[10] J. Duchi et al. "Efficient projections onto the l 1-ball for learning in high dimensions". In: *ICML*. ACM. 2008.

[11] D. V. Essen et al. "The Human Connectome Project: A data acquisition perspective". In: *NeuroImage* 62 (2012).

[12] K. J. Friston et al. "Statistical Parametric Maps in Functional Imaging: A General Linear Approach". In: *Hum Brain Mapp* (1995).

[13] L. Grosenick et al. "Interpretable whole-brain prediction analysis with GraphNet". In: *NeuroImage* 72 (2013).

[14] *HCP wiki*. `https://wiki.humanconnectome.org/display/PublicData/HCP+Data+Dictionary+Public-+500+Subject+Release`. Accessed: 2010-09-30.

[15] M. Hebiri and S. van de Geer. "The Smooth-Lasso and other ℓ_1 + ℓ_2-penalized methods". In: *Electron. J. Stat.* 5 (2011).

[16] D. P. Hibar et al. "Genetic clustering on the hippocampal surface for genome-wide association studies". In: *MICCAI*. 2013.

[17] R. Jenatton, G. Obozinski, and F. Bach. "Structured sparse principal component analysis". In: *AISTATS*. 2010.

[18] J. Mairal et al. "Online learning for matrix factorization and sparse coding". In: *Journal of Machine Learning Research* 11 (2010).

[19] A. Mensch et al. "Dictionary Learning for Massive Matrix Factorization". In: *ICML*. ACM. 2016.

[20] R. Saxe, M. Brett, and N. Kanwisher. "Divide and conquer: a defense of functional localizers". In: *Neuroimage* 30 (2006).

[21] S. M. Smith et al. "Advances in functional and structural MR image analysis and implementation as FSL". In: *Neuroimage* 23 (2004).

[22] E. Varol and C. Davatzikos. "Supervised block sparse dictionary learning for simultaneous clustering and classification in computational anatomy." eng. In: *Med Image Comput Comput Assist Interv* 17 (2014).

[23] G. Varoquaux et al. "A group model for stable multi-subject ICA on fMRI datasets". In: *Neuroimage* 51 (2010).

[24] G. Varoquaux et al. "Cohort-level brain mapping: learning cognitive atoms to single out specialized regions". In: *IPMI*. 2013.

[25] G. Varoquaux et al. "FAASTA: A fast solver for total-variation regularization of ill-conditioned problems with application to brain imaging". In: *arXiv:1512.06999* (2015).

[26] G. Varoquaux et al. "Multi-subject dictionary learning to segment an atlas of brain spontaneous activity". In: *Inf Proc Med Imag*. 2011.

[27] Y. Ying and D.-X. Zhou. "Online regularized classification algorithms". In: *IEEE Trans. Inf. Theory* 52 (2006).

A Convergence of the proposed algorithm

We now show how the convergence of our proposed algorithm follows effortless from [18]. Note the the objective in (2) can be rewritten as function of the dictionary alone like so

$$\mathbb{E}_{\mathbf{x}}(\ell(\mathbf{x},\mathbf{V}))+\gamma\sum_{j=1}^{k}\Omega_{\text{Lap}}(\mathbf{V}^j)\overset{\text{a.s}}{=}\lim_{t\to\infty}f_t(\mathbf{V})+\gamma\sum_{j=1}^{k}\Omega_{\text{Lap}}(\mathbf{V}^j),$$

where $\ell(\mathbf{x},\mathbf{V}):=\min_{\mathbf{u}\in\mathbb{R}^k}\frac{1}{2}\|\mathbf{x}-\mathbf{V}\mathbf{u}\|_2^2+\frac{1}{2}\alpha\|\mathbf{u}\|_2^2$, and $f_t(\mathbf{V}):=\frac{1}{t}\sum_{i=1}^{t}\ell(\mathbf{X}_i,\mathbf{V})$, with the $\mathbf{X}_i$'s sampled from the data. For each time $t\geq 0$, define

$$\hat{f}_t(\mathbf{V}):=\frac{1}{t}\sum_{i=1}^{t}\left(\frac{1}{2}\|\mathbf{X}_i-\mathbf{V}\hat{\mathbf{U}}_i\|_2^2+\frac{1}{2}\alpha\|\hat{\mathbf{U}}_i\|_2^2\right)+\gamma\sum_{j=1}^{k}\Omega_{\text{Lap}}(\mathbf{V}^j) \tag{10}$$

with each code $\hat{\mathbf{U}}_i$ is computed online by solving the Ridge problem (4). The following observations are immediate:

- Assumption **(A)** of [18], that the data distribution admits compact support, is automatically true for all MRI and PET data, because it is imposed by acquisition (for example, there is no data beyond the bounding-box).
- Assumption **(C)**, that the solution to the coding problem, which corresponds to (4) in our case, is unique and Lipschitz continuous w.r.t to the incoming data $\mathbf{X}_t$ is automatically satisfied for us since $\hat{\mathbf{U}}_t(\mathbf{X}_t)=(\mathbf{V}^T\mathbf{V}+\alpha\mathbf{I})^{-1}\mathbf{V}^T\mathbf{X}_t$, a linear transformation of $\mathbf{X}_t$.

As a consequence, we have the classical convergence guarantees as in in [18]:

- Almost-sure convergence of the dictionary: $\mathbf{V}(t)-\mathbf{V}(t-1)=\mathcal{O}(1/t)$ a.s. as $t\to\infty$.
- Almost-sure convergence of the risk:
  - $\hat{f}_t(\mathbf{V}(t))$ converges a.s. as $t\to\infty$.
  - $f_t(\mathbf{V}(t))-\hat{f}(\mathbf{V}(t))\to 0$ a.s. as $t\to\infty$.
  - $f_t(\mathbf{V}(t))$ converges a.s. as $t\to\infty$.

Finally, one notes that since constraints in problem (2) are separable in the dictionary atoms $\mathbf{V}^j$, the BCD dictionary-update algorithm Alg. 2 is guaranteed to converge to a global optimum. [6, 18].

B Working in frequency domain, when it is possible.

To close this section, let us point out a few special instances cases of problem (6), for peculiar choices of the constraint set Q. First note that the objective in problem (6) can be conveniently rewritten as

$$\begin{aligned}F_{\gamma\mathbf{A}_t[j,j]^{-1}}(\mathbf{v},\mathbf{V}^j+\mathbf{A}_t[j,j]^{-1}(\mathbf{V}\mathbf{A}^j-\mathbf{B}_t^j))&=\frac{1}{2}(\mathbf{v}-\tilde{\mathbf{V}}^j)^T(\mathbf{I}-\gamma\mathbf{A}_t[j,j]^{-1}\Delta)(\mathbf{v}-\tilde{\mathbf{V}}^j)\\&=\frac{1}{2}(\hat{\mathbf{v}}-\hat{\tilde{\mathbf{V}}}^j)^T(\mathbf{I}-\gamma\mathbf{A}_t[j,j]^{-1}\Delta)(\hat{\mathbf{v}}-\hat{\tilde{\mathbf{V}}}^j),\end{aligned} \tag{11}$$

with

$$\tilde{\mathbf{V}}^j:=(\mathbf{A}_t[j,j]\mathbf{I}-\gamma\Delta)^{-1}\left(\mathbf{V}^j+\mathbf{A}_t[j,j]^{-1}(\mathbf{V}\mathbf{A}^j-\mathbf{B}_t^j)\right). \tag{12}$$

We note that the matrix-inversion $(\mathbf{I}-\tilde{\gamma}\Delta)^{-1}$ that appears in the formula above is a Laplacian filter, and can be efficiently applied in closed-form (i.e non-iteratively) in the Fourier / frequency domain. Indeed, under periodic boundary conditions, the discrete Laplacian Δ is Block-Circulant with Circulant Blocks (BCCB) and so is diagonalizable in the Fourier domain. Precisely,

$$\Delta=\mathcal{F}^*\Lambda\mathcal{F} \tag{13}$$

where the complex orthonormal operator $\mathcal{F}$ represents the fast Discrete Fourier Transform (DFT), and Λ is diagonal matrix made p eigenvalues (including multiplicities) of the Laplace operator Δ, given by

$$\Lambda(\omega) := -\sum_{d=1}^{3}\left(2\sin\left(\frac{\omega_d\pi}{2n_d}\right)\right)^2 = -2\sum_{d=1}^{3}\left(1-\cos\left(\frac{\omega_d\pi}{n_d}\right)\right) \leq 0,$$
$$\text{for } \omega = (\omega_1,\omega_2,\omega_3) \in [\![0,n_1-1]\!]\times[\![0,n_2-1]\!]\times[\![0,n_3-1]\!].$$

We note that the spectral norm of Laplace operator in D dimensions (here $D=3$) is $\|\Delta\|_2 = \tilde{\gamma}_{\max}(-\Delta) = 2\times D\times(1+1) = 4D$.

Now, one can then harvest the closed-form solution

$$(\mathbf{I}-\tilde{\gamma}\Delta)^{-1}\mathbf{a} = (\mathcal{F}^{-1}(\mathbf{I}-\tilde{\gamma}\Lambda)^{-1}\mathcal{F})(\mathbf{a}) = \mathcal{F}^{-1}(\mathbf{s}), \tag{14}$$

where $\mathbf{s}\in\mathbb{R}^p$ is defined by $\mathbf{s}(\omega) := \dfrac{\hat{\mathbf{a}}(\omega)}{1-\tilde{\gamma}\hat{\Delta}(\omega)}$, with $\hat{\mathbf{a}} := \mathcal{F}(\mathbf{a})$. These DFT computations have complexity $\mathcal{O}(p\log p)$.

For applying the DFTs above, one can use the FFTW library for computing the forward and inverse Fourier transforms needed to apply the Laplacian filter (refer to paragraph B). FFTW is generally taught to be one of the fastest implementations of the FFT, yielding up to $3\times$ speedup against competing libraries like LAPACK.

Pure ℓ_2 constraint. Here, the constraint set $\mathcal{C}$ is an L2 ball (with radius $=1$, w.l.o.g) in $\mathbb{R}^2$. By the Rayleigh energy theorem (aka Parseval's identity for the DFT), one has

$$\|\hat{\mathbf{v}}\|^2 = p\|\mathbf{v}\|_2^2,\ \forall\mathbf{v}\in\mathbb{R}^p$$

and so problem (6) can be written as

$$\begin{aligned}\mathbf{V}^j &\leftarrow \operatorname{argmin}_{\mathbf{v}\in\mathbb{R}^p,\ \|\mathbf{v}\|_2^2\leq 1}\frac{1}{2}(\hat{\mathbf{v}}-\hat{\bar{\mathbf{V}}}^j)^*(\mathbf{I}-\gamma\mathbf{A}_t[j,j]^{-1}\Lambda)(\hat{\mathbf{v}}-\hat{\bar{\mathbf{V}}}^j)\\ &= \mathcal{F}^*\left(\operatorname{argmin}_{\hat{\mathbf{v}}\in\mathbb{C}^p,\ \|\hat{\mathbf{v}}\|_2^2\leq p}\frac{1}{2}(\hat{\mathbf{v}}-\hat{\bar{\mathbf{V}}}^j)^*(\mathbf{I}-\gamma\mathbf{A}_t[j,j]^{-1}\Lambda)(\hat{\mathbf{v}}-\hat{\bar{\mathbf{V}}}^j)\right)\\ &= \mathcal{F}^*\left(P_{\mathcal{E}}(\hat{\bar{\mathbf{V}}}^j)\right)\end{aligned} \tag{15}$$

where

$$\mathcal{E} := \left\{(\mathbf{I}-\gamma\mathbf{A}_t[j,j]^{-1}\Lambda)^{\frac{1}{2}}\hat{\mathbf{v}}\ \text{s.t}\ \hat{\mathbf{v}}\in\mathbb{C}^p,\ \|\hat{\mathbf{v}}\|_2^2\leq p\right\}, \tag{16}$$

a hyper-ellipsoid in standard position (i.e $\mathbf{0}$-centered and axes-aligned). Using elementary geometric arguments, one can show that the projection $P_{\mathcal{E}}(\hat{\bar{\mathbf{V}}}^j)$ can be computed efficiently using a kind of root-finding algorithm [8], and converges exponentially fast.

Non-negative Lasso. In case the constraint set $\mathcal{C}$ for the dictionary atoms is a simplex $\mathcal{S}_p(\tau)$, the simplex (see section 2), then the BCD update for the jth atom becomes

$$\begin{aligned}\mathbf{V}^j &\leftarrow \operatorname{argmin}_{\mathbf{v}\in\mathbb{R}^p,\ \mathbf{v}\geq 0,\ \mathbf{1}^T\mathbf{v}\leq 1}\frac{1}{2}(\hat{\mathbf{v}}-\hat{\bar{\mathbf{V}}}^j)^*(\mathbf{I}-\gamma\mathbf{A}_t[j,j]^{-1}\Lambda)(\hat{\mathbf{v}}-\hat{\bar{\mathbf{V}}}^j)\\ &= \mathcal{F}^*\left(\operatorname{argmin}_{\hat{\mathbf{v}}\in\mathbb{C}^p,\ -\mathcal{F}^*\hat{\mathbf{v}}\leq 0,\ \hat{\mathbf{1}}^T\hat{\mathbf{v}}\leq 1}\frac{1}{2}(\hat{\mathbf{v}}-\hat{\bar{\mathbf{V}}}^j)^*(\mathbf{I}-\gamma\mathbf{A}_t[j,j]^{-1}\Lambda)(\hat{\mathbf{v}}-\hat{\bar{\mathbf{V}}}^j)\right),\end{aligned} \tag{17}$$

which is a diagonal quadratic program with linear constraints, and can be effectively solved via the well-known simplex method, for example.

C Analytic upper bound for regularization parameter in sparse-coding

Let $\mathbf{V}$ be the current dictionary at time $t\geq 0$ (with the t subscript dropped for ease of notation), and consider the sparse-coding problem

$$\mathbf{u}_t \leftarrow \operatorname*{argmin}_{\mathbf{u}\in\mathbb{R}^k}\frac{1}{2}\|\mathbf{V}\mathbf{u}-\mathbf{X}_t\|_2^2+\alpha\|\mathbf{u}\|_1, \tag{18}$$

which is equivalent to problem (4), with the choice of penalty $\phi = \|.\|_1$ on the codes. Now, it follows from the Lasso theory that $\mathbf{u}_t$ above will be the zero vector if $\alpha \geq \alpha_{\max}(t)$, where

$$\alpha_{\max}(t) := \|\mathbf{V}^T \mathbf{X}_t\|_\infty = \max_{1 \leq j \leq p} |\langle \mathbf{V}^j, \mathbf{X}_t \rangle|$$

Now, it is clear that

$$\alpha_{\max}(t) \leq \max_{1 \leq j \leq p} \|\mathbf{V}^j\|_1 \|\mathbf{X}_t\|_\infty \leq \|\mathbf{X}_t\|_\infty \leq \sup_{t \geq 0} \|\mathbf{X}_t\|_\infty,$$

the first inequality being a consequence of the Cauchy-Schwarz and the second is due to the constraints on the dictionary atoms $\|\mathbf{V}^j\|_1 \leq 1$. Defining,

$$\|\mathbf{X}\|_{\infty,\infty} := \sup_{t \geq 0} \|\mathbf{X}_t\|_\infty := \sup_{t \geq 0} \max_{1 \leq j \leq p} |\mathbf{X}_t^j|,$$

we then obtain the rule:

$$\text{If } \alpha \geq \sup_{t \geq 0} \max_{1 \leq j \leq p} |\mathbf{X}_t^j|, \text{ then } \mathbf{u}_t = 0, \ \forall t \geq 0. \tag{19}$$

In particular, this means that a cross-validation procedure for selecting α only need to consider values in the range $0 \leq \alpha \leq \|\mathbf{X}\|_{\infty,\infty}$, which depends only on the input data matrix $\mathbf{X}$.