

---

# Multivariate Convolutional Sparse Coding for Electromagnetic Brain Signals

---

Tom Dupré La Tour<sup>\*1</sup>, Thomas Moreau<sup>\*2</sup>, Mainak Jas<sup>1</sup>, Alexandre Gramfort<sup>2</sup>

1: LTCI, Télécom ParisTech, Université Paris-Saclay, Paris, France

2: INRIA, Université Paris-Saclay, Saclay, France

\*: Both authors contributed equally.

## Abstract

Frequency-specific patterns of neural activity are traditionally interpreted as sustained rhythmic oscillations, and related to cognitive mechanisms such as attention, high level visual processing or motor control. While alpha waves (8–12 Hz) are known to closely resemble short sinusoids, and thus are revealed by Fourier analysis or wavelet transforms, there is an evolving debate that electromagnetic neural signals are composed of more complex waveforms that cannot be analyzed by linear filters and traditional signal representations. In this paper, we propose to learn dedicated representations of such recordings using a multivariate convolutional sparse coding (CSC) algorithm. Applied to electroencephalography (EEG) or magnetoencephalography (MEG) data, this method is able to learn not only prototypical temporal waveforms, but also associated spatial patterns so their origin can be localized in the brain. Our algorithm is based on alternated minimization and a greedy coordinate descent solver that leads to state-of-the-art running time on long time series. To demonstrate the implications of this method, we apply it to MEG data and show that it is able to recover biological artifacts. More remarkably, our approach also reveals the presence of non-sinusoidal mu-shaped patterns, along with their topographic maps related to the somatosensory cortex.

## 1 Introduction

Neural activity recorded via measurements of the electrical potential over the scalp by electroencephalography (EEG), or magnetic fields by magnetoencephalography (MEG), can be used to investigate human cognitive processes and certain pathologies. Such recordings consist of dozens to hundreds of simultaneously recorded signals, for durations going from minutes to hours. In order to describe and quantify neural activity in such multi-gigabyte data, it is classical to decompose the signal in predefined representations such as the Fourier or wavelet bases. It leads to canonical frequency bands such as theta (4–8 Hz), alpha (8–12 Hz), or beta (15–30 Hz) (Buzsaki, 2006), in which signal power can be quantified. While such linear analyses have had significant impact in neuroscience, there is now a debate regarding whether neural activity consists more of transient bursts of isolated events rather than rhythmically sustained oscillations (van Ede et al., 2018). To study the transient events and the morphology of the waveforms (Mazaheri and Jensen, 2008; Cole and Voytek, 2017), which matter in cognition and for our understanding of pathologies (Jones, 2016; Cole et al., 2017), there is a clear need to go beyond traditionally employed signal processing methodologies (Cole and Voytek, 2018). For instance, a classic Fourier analysis fails to distinguish alpha-rhythms from mu-rhythms, which have the same peak frequency at around 10 Hz, but whose waveforms are different (Cole and Voytek, 2017; Hari and Puce, 2017).

The key to many modern statistical analyses of complex data such as natural images, sounds or neural time series is the estimation of data-driven representations. Dictionary learning is one family

of techniques, which consists in learning atoms (or patterns) that offer sparse data approximations. When working with long signals in which events can happen at any instant, one idea is to learn *shift-invariant* atoms. They can offer better signal approximations than generic bases such as Fourier or wavelets, since they are not limited to narrow frequency bands. Multiple approaches have been proposed to solve this shift-invariant dictionary learning problem, such as MoTIF (Jost et al., 2006), the sliding window matching (Gips et al., 2017), the adaptive waveform learning (Hitziger et al., 2017), or the learning of recurrent waveform (Brockmeier and Príncipe, 2016), yet they all have several limitations, as discussed in Jas et al. (2017). A more popular approach, especially in image processing, is the convolutional sparse coding (CSC) model (Jas et al., 2017; Pachitariu et al., 2013; Kavukcuoglu et al., 2010; Zeiler et al., 2010; Heide et al., 2015; Wohlberg, 2016b; Šorel and Šroubek, 2016; Grosse et al., 2007; Mailhé et al., 2008). The idea is to cast the problem as an optimization problem, representing the signal as a sum of convolutions between atoms and activation signals.

The CSC approach has been quite successful in several fields such as computer vision (Kavukcuoglu et al., 2010; Zeiler et al., 2010; Heide et al., 2015; Wohlberg, 2016b; Šorel and Šroubek, 2016), biomedical imaging (Jas et al., 2017; Pachitariu et al., 2013), and audio signal processing (Grosse et al., 2007; Mailhé et al., 2008), yet it was essentially developed for univariate signals. Interestingly, images can be multivariate such as color or hyper-spectral images, yet most CSC methods only consider gray scale images. To the best of our knowledge, the only reference to multivariate CSC is Wohlberg (2016a), where the author proposes two models well suited for 3-channel images. In the case of EEG and MEG recordings, neural activity is instantaneously and linearly spread across channels, due to Maxwell’s equations (Hari and Puce, 2017). The same temporal patterns are reproduced on all channels with different intensities, which depend on each activity’s location in the brain. To exploit this property, we propose to use a rank-1 constraint on each multivariate atom. This idea has been mentioned in (Barthélemy et al., 2012, 2013), but was considered less flexible than the full-rank model. Moreover, their proposed optimization techniques are not specific to shift-invariant models, and not scalable to long signals. Multivariate shift-invariant rank-1 decomposition of EEG has also been considered with matching pursuit (Durka et al., 2005), but without learning the atoms, which are fixed Gabor filters.

**Contribution** In this study, we develop a multivariate model for CSC, using a rank-1 constraint on the atoms to account for the instantaneous spreading of an electromagnetic source over all the channels. We also propose efficient optimization strategies, namely a locally greedy coordinate descent (LGCD, Moreau et al. 2018), and precomputation steps for faster gradient computations. We provide multiple numerical evaluations of our method, which show the highly competitive running time on both univariate and multivariate models, even when working with hundreds of channels. We also demonstrate the estimation performance of the multivariate model by recovering patterns on low signal-to-noise ratio (SNR) data. Finally, we illustrate our method with atoms learned on multivariate MEG data, that thanks to the rank-1 model can be localized in the brain for clinical or cognitive neuroscience studies.

**Notation** A multivariate signal with  $T$  time points in  $\mathbb{R}^P$  is noted  $X \in \mathbb{R}^{P \times T}$ , while  $x \in \mathbb{R}^T$  is a univariate signal. We index time with brackets  $X[t] \in \mathbb{R}^P$ , while  $X_i \in \mathbb{R}^T$  is the channel  $i$  in  $X$ . For a vector  $v \in \mathbb{R}^P$  we define the  $\ell_q$  norm as  $\|v\|_q = (\sum_i |v_i|^q)^{1/q}$ , and for a multivariate signal  $X \in \mathbb{R}^{P \times T}$ , we define the time-wise  $\ell_q$  norm as  $\|X\|_q = (\sum_{t=1}^T \|X[t]\|_q^q)^{1/q}$ . The transpose of a matrix  $U$  is denoted by  $U^\top$ . For a multivariate signal  $X \in \mathbb{R}^{P \times T}$ ,  $X^\nabla$  is obtained by reversal of the temporal dimension, i.e.,  $X^\nabla[t] = X[T + 1 - t]$ . The convolution of two signals  $z \in \mathbb{R}^{T-L+1}$  and  $d \in \mathbb{R}^L$  is denoted by  $z * d \in \mathbb{R}^T$ . For  $D \in \mathbb{R}^{P \times L}$ ,  $z * D$  is obtained by convolving every row of  $D$  by  $z$ . For  $D' \in \mathbb{R}^{P \times L}$ ,  $D \tilde{*} D' \in \mathbb{R}^{2L-1}$  is obtained by summing the convolution between each row of  $D$  and  $D'$ :  $D \tilde{*} D' = \sum_{p=1}^P D_p * D'_p$ . We note  $[a, b]$  the set of real numbers between  $a$  and  $b$ , and  $\llbracket a, b \rrbracket$  the set of integers between  $a$  and  $b$ . We define  $\tilde{T}$  as  $T - L + 1$ .

## 2 Multivariate Convolutional Sparse Coding

In this section, we introduce the convolutional sparse coding (CSC) models used in this work. We focus on 1D-convolution, although these models can be naturally extended to higher order signals such as images by using the proper convolution operators.

**Univariate CSC** The CSC formulation adopted in this work follows the shift-invariant sparse coding (SISC) model from [Grosse et al. \(2007\)](#). It is defined as follows:

$$\begin{aligned} \min_{\{d_k\}_k, \{z_k^n\}_{k,n}} \quad & \sum_{n=1}^N \frac{1}{2} \left\| x^n - \sum_{k=1}^K z_k^n * d_k \right\|_2^2 + \lambda \sum_{k=1}^K \|z_k^n\|_1, \\ \text{s.t.} \quad & \|d_k\|_2^2 \leq 1 \text{ and } z_k^n \geq 0, \end{aligned} \quad (1)$$

where  $\{x^n\}_{n=1}^N \subset \mathbb{R}^T$  are  $N$  observed signals,  $\lambda > 0$  is the regularization parameter,  $\{d_k\}_{k=1}^K \subset \mathbb{R}^L$  are the  $K$  temporal atoms we aim to learn, and  $\{z_k^n\}_{k=1}^K \subset \mathbb{R}^{\tilde{T}}$  are  $K$  signals of activations, a.k.a. the code associated with  $x^n$ . This model assumes that the coding signals  $z_k^n$  are sparse, in the sense that only few entries are nonzero in each signal. In this work, we also assume that the entries of  $z_k^n$  are positive, which means that the temporal patterns are present each time with the same polarity.

**Multivariate CSC** The multivariate formulation uses an additional dimension on the signals and on the atoms, since the signal is recorded over  $P$  channels (mapping to space locations):

$$\begin{aligned} \min_{\{D_k\}_k, \{z_k^n\}_{k,n}} \quad & \sum_{n=1}^N \frac{1}{2} \left\| X^n - \sum_{k=1}^K z_k^n * D_k \right\|_2^2 + \lambda \sum_{k=1}^K \|z_k^n\|_1, \\ \text{s.t.} \quad & \|D_k\|_2^2 \leq 1 \text{ and } z_k^n \geq 0, \end{aligned} \quad (2)$$

where  $\{X^n\}_{n=1}^N \subset \mathbb{R}^{P \times T}$  are  $N$  observed multivariate signals,  $\{D_k\}_{k=1}^K \subset \mathbb{R}^{P \times L}$  are the spatio-temporal atoms, and  $\{z_k^n\}_{k=1}^K \subset \mathbb{R}^{\tilde{T}}$  are the sparse activations associated with  $X^n$ .

**Multivariate CSC with rank-1 constraint** This model is similar to the multivariate case but it adds a rank-1 constraint on the dictionary,  $D_k = u_k v_k^\top \in \mathbb{R}^{P \times L}$ , with  $u_k \in \mathbb{R}^P$  being the pattern over channels and  $v_k \in \mathbb{R}^L$  the pattern over time. The optimization problem boils down to:

$$\begin{aligned} \min_{\{u_k\}_k, \{v_k\}_k, \{z_k^n\}_{k,n}} \quad & \sum_{n=1}^N \frac{1}{2} \left\| X^n - \sum_{k=1}^K z_k^n * (u_k v_k^\top) \right\|_2^2 + \lambda \sum_{k=1}^K \|z_k^n\|_1, \\ \text{s.t.} \quad & \|u_k\|_2^2 \leq 1, \|v_k\|_2^2 \leq 1 \text{ and } z_k^n \geq 0. \end{aligned} \quad (3)$$

The rank-1 constraint is consistent with Maxwell's equations and the physical model of electrophysiological signals like EEG or MEG, where each source is linearly spread instantaneously over channels with a constant topographic map ([Hari and Puce, 2017](#)). Using this assumption, one aims to improve the estimation of patterns under the presence of independent noise over channels. Moreover, it can help separating overlapped sources which are inherently rank-1 but whose sum is generally of higher rank. Finally, as explained below, several computations can be factorized to speed up computations.

**Noise model** Note that our models use a Gaussian noise, whereas one can also use an alpha-stable noise distribution to better handle strong artifacts, as proposed by [Jas et al. \(2017\)](#). Importantly, our contribution is orthogonal to their work, and one can easily extend multivariate models to alpha-stable noise distributions, by using their EM algorithm and by updating the  $\ell_2$  loss into a weighted  $\ell_2$  loss in (3). Also, our experiments used artifact-free datasets, so the Gaussian noise model is appropriate.

### 3 Model estimation

Problems (1), (2) and (3) share the same structure. They are convex in each variable but not jointly convex. The resolution is done by using a block coordinate descent approach which minimizes alternately the objective function over one block of the variables. In this section, we describe this approach on the multivariate CSC with rank-1 constraint case (3), updating iteratively the activations  $z_k^n$ , the spatial patterns  $u_k$ , and the temporal pattern  $v_k$ .

#### 3.1 Z-step: solving for the activations

Given  $K$  fixed atoms  $D_k$  and a regularization parameter  $\lambda > 0$ , the Z-step aims to retrieve the  $NK$  activation signals  $z_k^n \in \mathbb{R}^{\tilde{T}}$  associated to the signals  $X^n \in \mathbb{R}^{P \times T}$  by solving the following

---

**Algorithm 1:** Locally greedy coordinate descent (LGCD)

---

**Input :** Signal  $X$ , atoms  $D_k$ , number of segments  $M$ , stopping parameter  $\epsilon > 0$ ,  $z_k$  initialization  
Initialize  $\beta_k[t]$  with (5).

**repeat**

**for**  $m = 1$  **to**  $M$  **do**

    Compute  $z'_k[t] = \max\left(\frac{\beta_k[t] - \lambda}{\|D_k\|_2^2}, 0\right)$  for  $(k, t) \in \mathcal{C}_m$

    Choose  $(k_0, t_0) = \arg \max_{(k,t) \in \mathcal{C}_m} |z_k[t] - z'_k[t]|$

    Update  $\beta$  with (6)

    Update the current point estimate  $z_{k_0}[t_0] \leftarrow z'_{k_0}[t_0]$

**until**  $\|z - z'\|_\infty < \epsilon$ 

---

$\ell_1$ -regularized optimization problem:

$$\min_{\substack{\{z_k^n\}_{k,n} \\ z_k^n \geq 0}} \frac{1}{2} \left\| X^n - \sum_{k=1}^K z_k^n * D_k \right\|_2^2 + \lambda \sum_{k=1}^K \|z_k^n\|_1. \quad (4)$$

This problem is convex in  $z_k^n$  and can be efficiently solved. In [Chalasani et al. \(2013\)](#), the authors proposed an algorithm based on FISTA ([Beck and Teboulle, 2009](#)) to solve it. [Bristow et al. \(2013\)](#) introduced a method based on ADMM ([Boyd et al., 2011](#)) to compute efficiently the activation signals  $z_k^n$ . These two methods are detailed and compared by [Wohlberg \(2016b\)](#), which also made use of the fast Fourier transform (FFT) to accelerate the computations. Recently, [Jas et al. \(2017\)](#) proposed to use L-BFGS ([Byrd et al., 1995](#)) to improve on first order methods. Finally, [Kavukcuoglu et al. \(2010\)](#) adapted the greedy coordinate descent (GCD) to solve this convolutional sparse coding problem.

However, for long signals, these techniques can be quite slow due the computation of the gradient (FISTA, ADMM, L-BFGS) or the choice of the best coordinate to update in GCD, which are operations that scale linearly in  $T$ . A way to alleviate this limitation is to use a locally greedy coordinate descent (LGCD) strategy, presented recently in [Moreau et al. \(2018\)](#).

Note that problem (4) is independent for each signal  $X^n$ . The computation of each  $z^n$  can thus be parallelized, independently of the technique selected to solve the optimization ([Jas et al., 2017](#)). Therefore, we omit the superscript  $n$  in the following subsection to simplify the notation.

**Coordinate descent (CD)** The key idea of coordinate descent is to update our estimate of the solution one coordinate  $z_k[t]$  at a time. For (4), it is possible to compute the optimal value  $z'_k[t]$  of one coordinate  $z_k[t]$  given that all the others are fixed. Indeed, the problem (4) restricted to one coordinate has a closed-form solution given by:

$$z'_k[t] = \max\left(\frac{\beta_k[t] - \lambda}{\|D_k\|_2^2}, 0\right), \text{ with } \beta_k[t] = \left[ D_k^\dagger \tilde{*} \left( X - \sum_{l=1}^K z_l * D_l + z_k[t] e_t * D_k \right) \right] [t] \quad (5)$$

where  $e_t \in \mathbb{R}^{\tilde{T}}$  is the canonical basis vector with value 1 at index  $t$  and 0 elsewhere. When updating the coefficient  $z_{k_0}[t_0]$  to the value  $z'_{k_0}[t_0]$ ,  $\beta$  is updated with:

$$\beta_k^{(q+1)}[t] = \beta_k^{(q)}[t] + (D_{k_0}^\dagger \tilde{*} D_k)[t - t_0](z_{k_0}[t_0] - z'_{k_0}[t_0]), \quad \forall (k, t) \neq (k_0, t_0). \quad (6)$$

The term  $(D_{k_0}^\dagger \tilde{*} D_k)[t - t_0]$  is zero for  $|t - t_0| \geq L$ . Thus, only  $K(2L - 1)$  coefficients of  $\beta$  need to be changed ([Kavukcuoglu et al., 2010](#)). The CD algorithm updates at each iteration a coordinate to this optimal value. The coordinate to update can be chosen with different strategies, such as the cyclic strategy which iterates over all coordinates ([Friedman et al., 2007](#)), the randomized CD ([Nesterov, 2010](#); [Richtárik and Takáč, 2014](#)) which chooses a coordinate at random for each iteration, or the greedy CD ([Osher and Li, 2009](#)) which chooses the coordinate the farthest from its optimal value.

**Locally greedy coordinate descent (LGCD)** The choice of a coordinate selection strategy results of a tradeoff between the computational cost of each iteration and the improvement it provides. For cyclic and randomized strategies, the iteration complexity is  $\mathcal{O}(KL)$  as the coordinate selection can

Table 1: Computational complexities of each step

Step	Computation	Computed	Rank-1	Full-rank
Z-step	$\beta$ initialization	once	$NKT(L + P)$	$NKT(LP)$
Z-step	Precomputation	once	$K^2L(L + P)$	$K^2L(LP)$
Z-step	M coordinate updates	multiple times	$MKL$	$MKL$
D-step	$\Phi$ precomputation	once	$NKTLP$	$NKTLP$
D-step	$\Psi$ precomputation	once	$NK^2TL$	$NKTLP$
D-step	Gradient evaluation	multiple times	$K^2L(L + P)$	$K^2L(LP)$
D-step	Function evaluation	multiple times	$K^2L(L + P)$	$K^2L(LP)$

be performed in constant time. The greedy selection of a coordinate is more expensive as it is linear in the signal length  $\mathcal{O}(K\tilde{T})$ . However, greedy selection is more efficient iteration-wise (Nutini et al., 2015). Moreau et al. (2018) proposed to consider a locally greedy selection strategy for CD. The coordinate to update is chosen greedily in one of  $M$  subsegments of the signal, *i.e.*, at iteration  $q$ , the selected coordinate is:

$$(k_0, t_0) = \arg \max_{(k,t) \in \mathcal{C}_m} |z_k[t] - z'_k[t]|, \quad m \equiv q \pmod{M} + 1, \quad (7)$$

with  $\mathcal{C}_m = \llbracket 1, K \rrbracket \times \llbracket (m-1)\tilde{T}/M, m\tilde{T}/M \rrbracket$ . With this strategy, the coordinate selection complexity is linear in the length of the considered subsegment  $\mathcal{O}(K\tilde{T}/M)$ . By choosing  $M = \lfloor \tilde{T}/(2L-1) \rfloor$ , the complexity of update is the same as the complexity of random and cyclic coordinate selection,  $\mathcal{O}(KL)$ . We detail the steps of LGCD in Algorithm 1. This algorithm is particularly efficient when the  $z_k$  are sparser. Indeed, in this case, only few coefficients need to be updated in the signal, resulting in a low number of iterations. Computational complexities are detailed in Table 1.

**Relation with matching pursuit (MP)** Note that the greedy CD is strongly related to the well-known matching pursuit (MP) algorithm (Locatello et al., 2018). The main difference is that MP solves a slightly different problem, where the  $\ell_1$  regularization is replaced with an  $\ell_0$  constraint. Therefore, the size of the support is a fixed parameter in MP, whereas it is controlled by the regularization parameter  $\lambda$  in our case. In term of algorithm, both methods update one coordinate at a time selected greedily, but MP does not apply a soft-thresholding in (5).

### 3.2 D-step: solving for the atoms

Given  $KN$  fixed activation signals  $z_k^n \in \mathbb{R}^{\tilde{T}}$ , associated to signals  $X^n \in \mathbb{R}^{P \times T}$ , the  $D$ -step aims to update the  $K$  spatial patterns  $u_k \in \mathbb{R}^P$  and  $K$  temporal patterns  $v_k \in \mathbb{R}^L$ , by solving:

$$\min_{\substack{\|u_k\|_2 \leq 1 \\ \|v_k\|_2 \leq 1}} E, \quad \text{where} \quad E \triangleq \sum_{n=1}^N \frac{1}{2} \|X^n - \sum_{k=1}^K z_k^n * (u_k v_k^\top)\|_2^2. \quad (8)$$

The problem (8) is convex in each block of variables  $\{u_k\}_k$  and  $\{v_k\}_k$ , but not jointly convex. Therefore, we optimize first  $\{u_k\}_k$ , then  $\{v_k\}_k$ , using in both cases a projected gradient descent with an Armijo backtracking line-search (Wright and Nocedal, 1999) to find a good step size. These steps are detailed in Algorithm A.1.

**Gradient relative to  $u_k$  and  $v_k$**  The gradient of  $E$  relatively to  $\{u_k\}_k$  and  $\{v_k\}_k$  can be computed using the chain rule. First, we compute the gradient relatively to a full atom  $D_k = u_k v_k^\top \in \mathbb{R}^{P \times L}$ :

$$\nabla_{D_k} E = \sum_{n=1}^N (z_k^n)^\top * \left( X^n - \sum_{l=1}^K z_l^n * D_l \right) = \Phi_k - \sum_{l=1}^K \Psi_{k,l} * D_l, \quad (9)$$

where we reordered this expression to define  $\Phi_k \in \mathbb{R}^{P \times L}$  and  $\Psi_{k,l} \in \mathbb{R}^{2L-1}$ . These terms are both constant during a  $D$ -step and can thus be precomputed to accelerate the computation of the

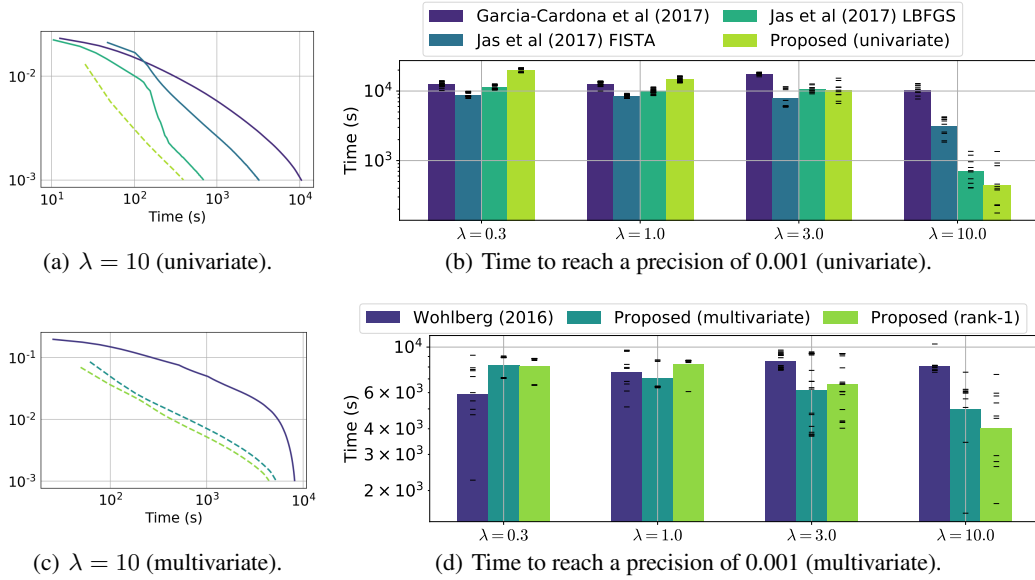


Figure 1: Comparison of state-of-the-art univariate (a, b) and multivariate (c, d) methods with our approach. (a) Convergence plot with the objective function relative to the obtained minimum, as a function of computational time. (b) Time taken to reach a relative precision of  $10^{-3}$ , for different regularization parameters  $\lambda$ . (c, d) Same as (a, b) in the multivariate setting  $P = 5$ .

gradients and the cost function  $E$ . We detail these computations in the supplementary materials (see Section A.1). Computational complexities are detailed in Table 1. Note that the dependence in  $T$  is present *only* in the precomputations, which makes the following iterations very fast. Without precomputations, the complexity of *each* gradient computation in the  $D$ -step would be  $\mathcal{O}(NKTLP)$ .

### 3.3 Initialization

The activations sub-problem ( $Z$ -step) is regularized with a  $\ell_1$ -norm, which induces sparsity: the higher the regularization parameter  $\lambda$ , the higher the sparsity. Therefore, there exists a value  $\lambda_{max}$  above which the sub-problem solution is always zeros (Hastie et al., 2015). As  $\lambda_{max}$  depends on the atoms  $D_k$  and on the signals  $X^n$ , its value changes after each  $D$ -step. In particular, its value might change a lot between the initialization and the first  $D$ -step. This is problematic since we cannot use a regularization  $\lambda$  above this initial  $\lambda_{max}$ , even though the following  $\lambda_{max}$  might be higher.

The standard strategy to initialize CSC methods is to generate random atoms with Gaussian white noise. However, as these atoms generally poorly correlate with the signals, the initial value of  $\lambda_{max}$  is low compared to the following ones. For example, on the MEG dataset described later on, we found that the initial  $\lambda_{max}$  is about 1/3 of the following ones in the univariate case, with  $L = 32$ . On the multivariate case, it is even more problematic as with  $P = 204$ , we could have an initial  $\lambda_{max}$  as low as 1/20 of the following ones.

To fix this problem, we propose to initialize the dictionary with random chunks of the signal, projecting each chunk on a rank-1 approximation using singular value decomposition. We noticed on the MEG dataset that the initial  $\lambda_{max}$  was then about the same value as the following ones, which enables the use of higher regularization parameters. We used this scheme in all our experiments.

## 4 Experiments

All numerical experiments were run using Python (Python Software Foundation, 2017) and our code is publicly available online at <https://alphacsc.github.io/>.

**Speed performance** To illustrate the performance of our optimization strategy, we monitored its convergence speed on a real MEG dataset. The somatosensory dataset from the MNE software (Gram-



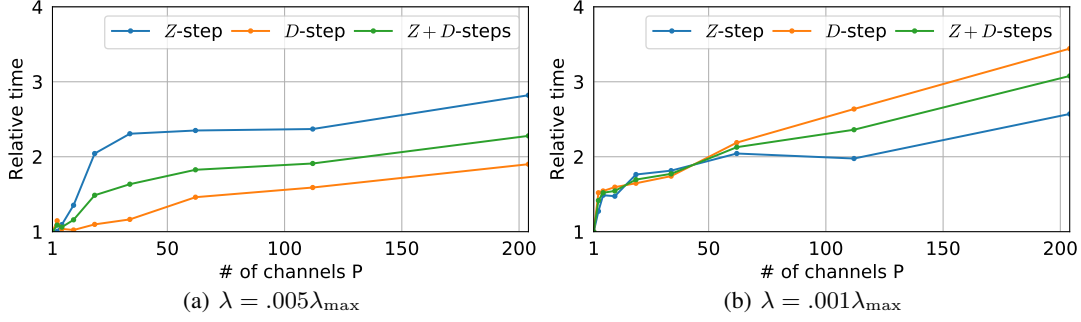


Figure 2: Timings of  $Z$  and  $D$  updates when varying the number of channels  $P$ . The scaling is sublinear with  $P$ , due to the precomputation steps in the optimization.

fort et al., 2013, 2014) contains responses to median nerve stimulation. We consider only gradiometers channels and we used the following parameters:  $T = 134\,700$ ,  $N = 2$ ,  $K = 8$ , and  $L = 128$ .

First we compared our strategy against three state-of-the-art *univariate* CSC solvers available online. The first was developed by Garcia-Cardona and Wohlberg (2017) and is based on ADMM. The second and third were developed by Jas et al. (2017), and are respectively based on FISTA and L-BFGS. All solvers shared the same objective function, but as the problem is non-convex, the solvers are not guaranteed to reach the same local minima, even though we started from the same initial settings. Hence, for a fair comparison, we computed the convergence curves relative to each local minimum, and averaged them over 10 different initializations. The results, presented in Figure 1(a, b), demonstrate the competitiveness of our method, for reasonable choices of  $\lambda$ . Indeed, a higher regularization parameter leads to sparser activations  $z_k^n$ , on which LGCD is particularly efficient.

Then, we also compared our method against a multivariate ADMM solver developed by Wohlberg (2016a). As this solver was quite slow on these long signals, we limited our experiments to  $P = 5$  channels. The results, presented in Figure 1(c, d), show that our method is faster than the competing method for large  $\lambda$ . More benchmarks are available in the supplementary materials.

**Scaling with the number of channels** The multivariate model involves an extra dimension  $P$  but its impact on the computational complexity of our solver is limited. Figure 2 shows the average running times of the  $Z$ -step and the  $D$ -step. Timings are normalized *w.r.t.* the timings for a single channel. The running times are computed using the same signals from the somatosensory dataset, with the following parameters:  $T = 26\,940$ ,  $N = 10$ ,  $K = 2$ ,  $L = 128$ . We can see that the scaling of these three operations is sub-linear in  $P$ . For the  $Z$ -step, only the initial computations for the first  $\beta_k$  and the constants  $D_k^{\dagger} \tilde{*} D_l$  depend linearly on  $P$  so that the complexity increase is limited compared to the complexity of solving the optimization problem (4). For the  $D$ -step, the scaling to compute the gradients is linear with  $P$ . However, the most expensive operations here are the computation of the constant  $\Psi_k$ , which does not on  $P$ .

**Finding patterns in low SNR signals** Since the multivariate model has access to more data, we would expect it to perform better compared to the univariate model especially for low SNR signals. To demonstrate this, we compare the two models when varying the number of channels  $P$  and the SNR of the data. The original dictionary contains two temporal patterns, a square and a triangle, presented in Figure 3(a). The spatial maps are designed with a sine and a cosine, and the first channel’s amplitude is forced to 1 to make sure both atoms are present even with only one channel. The signals are obtained by convolving the atoms with activation signals  $z_k^n$ , where the activation locations are sampled uniformly in  $\llbracket 1, T \rrbracket \times \llbracket 1, K \rrbracket$  with 5% non-zero activations, and the amplitudes are uniformly sampled in  $[0, 1]$ . Then, a Gaussian white noise with variance  $\sigma$  is added to the signal. We fixed  $N = 100$ ,  $L = 64$  and  $\tilde{T} = 640$  for our simulated signals. We can see in Figure 3(a) the temporal patterns recovered for  $\sigma = 10^{-3}$  using only one channel and using 5 channels. While the patterns recovered with one channel are very noisy, the multivariate model with rank-1 constraint recovers the original atoms accurately. This can be expected as the univariate model is ill-defined in this situation, where some atoms are superimposed. For the rank-1 model, as the atoms have different spatial maps, the problem is easier. Then, we evaluate the learned temporal atoms. Due to permutation and sign ambiguity, we compute the  $\ell_2$ -norm of the difference between the temporal

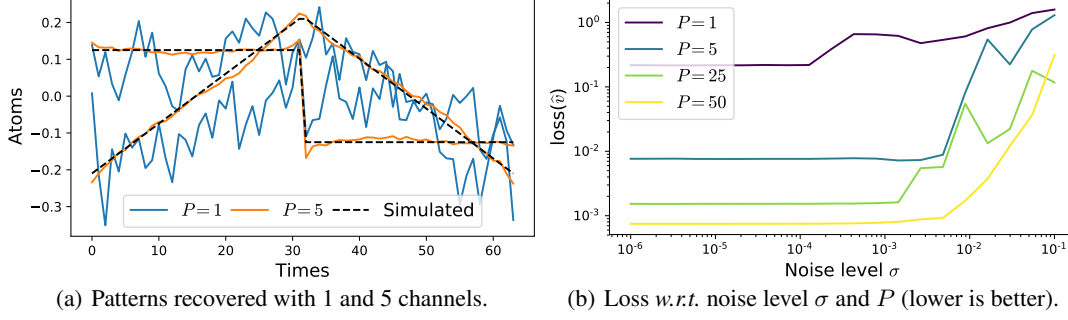


Figure 3: (a) Patterns recovered with  $P = 1$  and  $P = 5$ . The signals were generated with the two simulated temporal patterns and with  $\sigma = 10^{-3}$ . (b) Evolution of the recovery loss with  $\sigma$  for different values of  $P$ . Using more channels improves the recovery of the original patterns.

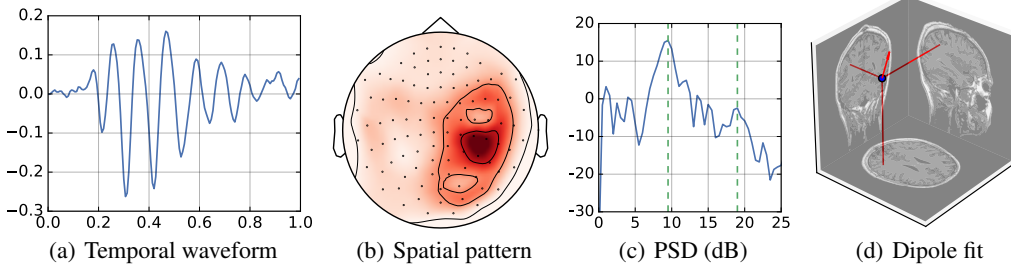


Figure 4: Atom revealed using the MNE somatosensory data. Note the non-sinusoidal comb shape of the mu rhythm. This atom has been manually selected, and other atoms are presented in Figure B.4.

pattern  $\hat{v}_k$  and the ground truths,  $v_k$  or  $-v_k$ , for all permutations  $\mathfrak{S}(K)$  i.e.,

$$\text{loss}(\hat{v}) = \min_{s \in \mathfrak{S}(K)} \sum_{k=1}^K \min \left( \|\hat{v}_{s(k)} - v_k\|_2^2, \|\hat{v}_{s(k)} + v_k\|_2^2 \right). \quad (10)$$

Multiple values of  $\lambda$  were tested and the best loss is reported in Figure 3(b) for varying noise levels  $\sigma$ . We observe that independently of the noise level, the multivariate rank-1 model outperforms the univariate one. This is true even for good SNR, as using multiple channels disambiguates the separation of overlapped patterns.

**Examples of atoms in real MEG signals:** We show the results of our algorithm on experimental data, using the MNE somatosensory dataset (Gramfort et al., 2013, 2014). This dataset contains MEG recordings of one patient receiving median nerve stimulations. Here we first extract  $N = 103$  trials from the data. Each trial lasts 6 s with a sampling frequency of 150 Hz ( $T = 900$ ). We selected only gradiometer channels, leading to  $P = 204$  channels. The signals were notch-filtered to remove the power-line noise, and high-pass filtered at 2 Hz to remove the low-frequency trend, i.e. to remove low frequency drift artifacts which contribute a lot to the variance of the raw signals. We learned  $K = 40$  atoms with  $L = 150$  using a rank-1 multivariate CSC model, with a regularization  $\lambda = 0.2\lambda_{max}$ .

Figure 4(a) shows a recovered non-sinusoidal brain rhythm which resembles the well-known mu-rhythm. The mu-rhythm has been implicated in motor-related activity (Hari, 2006) and is centered around 9–11 Hz. Indeed, while the power is concentrated in the same frequency band as the alpha, it has a very different spatial topography (Figure 4(b)). In Figure 4(c), the power spectral density (PSD) shows two components of the mu-rhythm – one at around 9 Hz, and a harmonic at 18 Hz as previously reported in (Hari, 2006). Based on our analysis, it is clear that the 18 Hz component is simply a harmonic of the mu-rhythm even though a Fourier-based analysis could lead us to falsely conclude that the data contained beta-rhythms. Finally, due to the rank-1 nature of our atoms, it is straightforward to fit an equivalent current dipole (Tuomisto et al., 1983) to interpret the origin of the signal. Figure 4(d) shows that the atom does indeed localize in the primary somatosensory cortex, or the so-called S1 region with a 59.3% goodness of fit. For results on more MEG datasets, see Section B.2. It notably includes mu-shaped atoms from S2.



## 5 Conclusion

Many neuroscientific debates today are centered around the morphology of the signals under consideration. For instance, are alpha-rhythms asymmetric (Mazaheri and Jensen, 2008) ? Are frequency specific patterns the result of sustained oscillations or transient bursts (van Ede et al., 2018) ? In this paper, we presented a multivariate extension to the CSC problem applied to MEG data to help answer such questions. In the original CSC formulation, the signal is expressed as a convolution of atoms and their activations. Our method extends this to the case of multiple channels and imposes a rank-1 constraint on the atoms to account for the instantaneous propagation of electromagnetic fields. We demonstrate the usefulness of our method on publicly available multivariate MEG data. Not only are we able to recover neurologically plausible atoms, but also we are able to find temporal waveforms which are non-sinusoidal. Empirical evaluations show that our solvers are significantly faster compared to existing CSC methods even for the univariate case (single channel). The algorithm scales sublinearly with the number of channels which means it can be employed even for dense sensor arrays with 200-300 sensors, leading to better estimation of the patterns and their origin in the brain.

## Acknowledgment

This work was supported by the ERC Starting Grant SLAB ERC-YStG-676943 and by the ANR THALAMEEG ANR-14-NEUC-0002-01.

## References

- Q. Barthélemy, A. Larue, A. Mayoue, D. Mercier, and J. I. Mars. Shift & 2D rotation invariant sparse coding for multivariate signals. *IEEE Transactions on Signal Processing*, 60(4):1597–1611, 2012.
- Q. Barthélemy, C. Gouy-Pailler, Y. Isaac, A. Souloumiac, A. Larue, and J. I. Mars. Multivariate temporal dictionary learning for EEG. *J. Neurosci. Methods*, 215(1):19–28, 2013.
- A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.
- S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.
- H. Bristow, A. Eriksson, and S. Lucey. Fast convolutional sparse coding. In *Computer Vision and Pattern Recognition (CVPR)*, pages 391–398, 2013.
- A. J. Brockmeier and J. C. Principe. Learning recurrent waveforms within EEGs. *IEEE Transactions on Biomedical Engineering*, 63(1):43–54, 2016.
- G. Buzsaki. *Rhythms of the Brain*. Oxford University Press, 2006.
- R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995.
- R. Chalasani, J. C. Principe, and N. Ramakrishnan. A fast proximal method for convolutional sparse coding. In *International Joint Conference on Neural Networks (IJCNN)*, pages 1–5, 2013. ISBN 9781467361293.
- S. R. Cole and B. Voytek. Brain oscillations and the importance of waveform shape. *Trends Cogn. Sci.*, 2017.
- S. R. Cole and B. Voytek. Cycle-by-cycle analysis of neural oscillations. *preprint bioRxiv*, 2018.
- S. R. Cole, R. van der Meij, E. J. Peterson, C. de Hemptinne, P. A. Starr, and B. Voytek. Nonsinusoidal beta oscillations reflect cortical pathophysiology in Parkinson’s disease. *Journal of Neuroscience*, 37(18):4830–4840, 2017.
- P. J. Durka, A. Matysiak, E. M. Montes, P. V. Sosa, and K. J. Blinowska. Multichannel matching pursuit and EEG inverse solutions. *J. Neurosci. Methods*, 148(1):49–59, 2005.

- J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani. Pathwise coordinate optimization. *The Annals of Applied Statistics*, 1(2):302–332, 2007.
- C. Garcia-Cardona and B. Wohlberg. Convolutional dictionary learning. *arXiv preprint arXiv:1709.02893*, 2017.
- B. Gips, A. Bahramisharif, E. Lowet, M. Roberts, P. de Weerd, O. Jensen, and J. van der Eerden. Discovering recurring patterns in electrophysiological recordings. *J. Neurosci. Methods*, 275: 66–79, 2017.
- A. Gramfort, M. Luessi, E. Larson, D. A. Engemann, D. Strohmeier, C. Brodbeck, R. Goj, M. Jas, T. Brooks, L. Parkkonen, and M. S. Hämäläinen. MEG and EEG data analysis with MNE-Python. *Frontiers in neuroscience*, 7, 2013.
- A. Gramfort, M. Luessi, E. Larson, D. A. Engemann, D. Strohmeier, C. Brodbeck, L. Parkkonen, and M. S. Hämäläinen. MNE software for processing MEG and EEG data. *Neuroimage*, 86:446–460, 2014.
- R. Grosse, R. Raina, H. Kwong, and A. Y. Ng. Shift-invariant sparse coding for audio classification. In *23rd Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 149–158. AUAI Press, 2007. ISBN 0-9749039-3-0.
- R. Hari. Action–perception connection and the cortical mu rhythm. *Progress in brain research*, 159: 253–260, 2006.
- R. Hari and A. Puce. *MEG-EEG Primer*. Oxford University Press, 2017.
- T. Hastie, R. Tibshirani, and M. J. Wainwright. *Statistical Learning with Sparsity*. CRC Press, 2015.
- F. Heide, W. Heidrich, and G. Wetzstein. Fast and flexible convolutional sparse coding. In *Computer Vision and Pattern Recognition (CVPR)*, pages 5135–5143. IEEE, 2015.
- S. Hitziger, M. Clerc, S. Sallet, C. Benar, and T. Papadopoulos. Adaptive Waveform Learning: A Framework for Modeling Variability in Neurophysiological Signals. *IEEE Transactions on Signal Processing*, 2017.
- M. Jas, T. Dupré La Tour, U. Şimşekli, and A. Gramfort. Learning the morphology of brain signals using alpha-stable convolutional sparse coding. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1–15, 2017.
- S. R. Jones. When brain rhythms aren’t ‘rhythmic’: implication for their mechanisms and meaning. *Curr. Opin. Neurobiol.*, 40:72–80, 2016.
- P. Jost, P. Vanderghenst, S. Lesage, and R. Gribonval. MoTIF: an efficient algorithm for learning translation invariant dictionaries. In *Acoustics, Speech and Signal Processing (ICASSP)*, volume 5. IEEE, 2006.
- K. Kavukcuoglu, P. Sermanet, Y.-L. Boureau, K. Gregor, M. Mathieu, and Y. Le Cun. Learning convolutional feature hierarchies for visual recognition. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1090–1098, 2010.
- F. Locatello, A. Raj, S. P. Karimireddy, G. Rätsch, B. Schölkopf, S. Stich, and M. Jaggi. On matching pursuit and coordinate descent. In *International Conference on Machine Learning (ICML)*, pages 3204–3213, 2018.
- B. Mailhé, S. Lesage, R. Gribonval, F. Bimbot, and P. Vanderghenst. Shift-invariant dictionary learning for sparse representations: extending K-SVD. In *16th Eur. Signal Process. Conf.*, pages 1–5. IEEE, 2008.
- A. Mazaheri and O. Jensen. Asymmetric amplitude modulations of brain oscillations generate slow evoked responses. *The Journal of Neuroscience*, 28(31):7781–7787, 2008.
- T. Moreau, L. Oudre, and N. Vayatis. DICOD: Distributed Convolutional Sparse Coding. In *International Conference on Machine Learning (ICML)*, 2018.

- Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2010.
- J. Nutini, M. Schmidt, I. H. Laradji, M. P. Friedlander, and H. Koepke. Coordinate Descent Converges Faster with the Gauss-Southwell Rule Than Random Selection. In *International Conference on Machine Learning (ICML)*, pages 1632–1641, 2015.
- S. Osher and Y. Li. Coordinate descent optimization for  $\ell_1$  minimization with application to compressed sensing; a greedy algorithm. *Inverse Problems and Imaging*, 3(3):487–503, 2009.
- M. Pachitariu, A. M Packer, N. Pettit, H. Dalgleish, M. Hausser, and M. Sahani. Extracting regions of interest from biological images with convolutional sparse block coding. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1745–1753, 2013.
- Python Software Foundation. Python Language Reference, version 3.6. <http://python.org/>, 2017.
- P. Richtárik and M. Takáč. Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Mathematical Programming*, 144(1-2):1–38, 2014.
- M. Šorel and F. Šroubek. Fast convolutional sparse coding using matrix inversion lemma. *Digital Signal Processing*, 2016.
- T. Tuomisto, R. Hari, T. Katila, T. Poutanen, and T. Varpula. Studies of auditory evoked magnetic and electric responses: Modality specificity and modelling. *Il Nuovo Cimento D*, 2(2):471–483, 1983.
- F. van Ede, A. J. Quinn, M. W. Woolrich, and A. C. Nobre. Neural oscillations: Sustained rhythms or transient burst-events? *Trends in Neurosciences*, 2018.
- B. Wohlberg. Convolutional sparse representation of color images. In *IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI)*, pages 57–60, 2016a.
- B. Wohlberg. Efficient algorithms for convolutional sparse representations. *Image Processing, IEEE Transactions on*, 25(1):301–315, 2016b.
- S. Wright and J. Nocedal. *Numerical optimization*, volume 35. Springer Science, 1999.
- M. D. Zeiler, D. Krishnan, G.W. Taylor, and R. Fergus. Deconvolutional networks. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2528–2535. IEEE, 2010.

## A Optimization details

In this section, we give more details about the optimization procedures used to speed-up both  $D$ -step and  $Z$ -step.

### A.1 Details on the $D$ -step optimization

First, let's recall the objective function, as introduced in [Section 3.2](#):

$$E \triangleq \sum_{n=1}^N \frac{1}{2} \|X^n - \sum_{k=1}^K z_k^n * (u_k v_k^\top)\|_2^2, \quad (\text{A.1})$$

which we optimize under the constraints  $\|u_k\|_2^2 \leq 1$  and  $\|v_k\|_2^2 \leq 1$ .

To compute the gradient of  $E$  relatively to a full atom  $D_k = u_k v_k^\top \in \mathbb{R}^{P \times L}$ , we introduce some constants  $\Phi_k$  and  $\Psi_{k,l}$ , which are constant during the entire  $D$ -step:

$$\nabla_{D_k} E = \sum_{n=1}^N (z_k^n)^\dagger * \left( X^n - \sum_{l=1}^K z_l^n * D_l \right) = \Phi_k - \sum_{l=1}^K \Psi_{k,l} * D_l \quad (\text{A.2})$$

Indeed, we have:

$$\nabla_{D_k} E[t] = \sum_{n=1}^N \left( (z_k^n)^\dagger * \left( X^n - \sum_{l=1}^K z_l^n * D_l \right) \right) [t] \quad (\text{A.3})$$

$$= \sum_{n=1}^N \sum_{\tau=1}^{\tilde{T}} z_k^n[\tau] \left( X^n - \sum_{l=1}^K z_l^n * D_l \right) [t + \tau - 1] \quad (\text{A.4})$$

$$= \sum_{n=1}^N \sum_{\tau=1}^{\tilde{T}} z_k^n[\tau] \left( X^n[t + \tau - 1] - \sum_{l=1}^K \sum_{\tau'=1}^L z_l^n[\tau'] D_l[t + \tau - \tau'] \right) \quad (\text{A.5})$$

$$= \Phi_k[t] - \sum_{l=1}^K \sum_{\tau'=1}^L \left( \sum_{n=1}^N \sum_{\tau=1}^{\tilde{T}} z_k^n[\tau] z_l^n[t + \tau - \tau'] \right) D_l[\tau'] \quad (\text{A.6})$$

$$= \Phi_k[t] - \sum_{l=1}^K \sum_{\tau'=1}^L \Psi_{k,l}[t + 1 - \tau'] D_l[\tau'] \quad (\text{A.7})$$

$$= \Phi_k[t] - \sum_{l=1}^K (\Psi_{k,l} * D_l)[t] \quad (\text{A.8})$$

where  $\Phi_k \in \mathbb{R}^{P \times L}$  are computed with:

$$\Phi_k[t] = \sum_{n=1}^N \sum_{\tau=1}^{\tilde{T}} z_k^n[\tau] X^n[t + \tau - 1], \quad \forall t \in \llbracket 1, L \rrbracket, \quad (\text{A.9})$$

and where  $\Psi_{k,l} \in \mathbb{R}^{2L-1}$  are computed with:

$$\Psi_{k,l}[t] = \sum_{n=1}^N \sum_{\tau=1}^{\tilde{T}} z_k^n[\tau] z_l^n[t + \tau - 1], \quad \forall t \in \llbracket 1, 2L - 1 \rrbracket. \quad (\text{A.10})$$

Note that in the last equation (A.10), the sum only concerns the defined terms, i.e.,  $(t + \tau - 1) \in \llbracket 1, \tilde{T} \rrbracket$ . The computational complexities of  $\Phi_k$  and  $\Psi_{k,l}$  are respectively  $\mathcal{O}(NLT KP)$  and  $\mathcal{O}(NLT K^2)$ .

Then, the gradients relative to  $u_k$  and  $v_k$  are obtained using the chain rule,

$$\nabla_{u_k} E = (\nabla_{D_k} E) v_k \in \mathbb{R}^P, \quad (\text{A.11})$$

$$\nabla_{v_k} E = u_k^\top (\nabla_{D_k} E) \in \mathbb{R}^L, \quad (\text{A.12})$$

---

**Algorithm A.1:** Projected gradient descent for updating  $\{u_k\}$  and  $\{v_k\}$ .

---

**Input :** Signals  $\{X^n\}_n$ , activations  $\{z_k^n\}_{k,n}$ , stopping parameter  $\epsilon > 0$ ,  
initial estimate  $\{u_k\}_k$  and  $\{v_k\}_k$

Initialize  $\Phi_k$  with (A.9) and  $\Psi_{k,l}$  with (A.10), for  $k, l \in \llbracket 1, K \rrbracket$ .

**repeat**

    Compute with (A.11) for  $k \in \llbracket 1, K \rrbracket$ ,  $G_k = \nabla_{u_k} E$ ,

    Update the estimate with  $\{u_k^{(q+1)}\}_k \leftarrow \text{to Armijo}(\{u_k^{(q)}\}_k, G_k, E)$

**until**  $\sum_{k=1}^K \left\| u_k^{(q+1)} - u_k^{(q)} \right\|_1 < \epsilon$

Set  $\{u_k\}_k \leftarrow \{u_k^{(q)}\}_k$

**repeat**

    Compute with (A.12) for  $k \in \llbracket 1, K \rrbracket$ ,  $G_k = \nabla_{v_k} E$ ,

    Update the estimate with  $\{v_k^{(q+1)}\}_k \leftarrow \text{to Armijo}(\{v_k^{(q)}\}_k, G_k, E)$

**until**  $\sum_{k=1}^K \left\| v_k^{(q+1)} - v_k^{(q)} \right\|_1 < \epsilon$

Set  $\{v_k\}_k \leftarrow \{v_k^{(q)}\}_k$

**return**  $\{u_k\}_k$  and  $\{v_k\}_k$

---

and  $E$  can be computed, up to a constant term  $C$ , with the following

$$E = \sum_{k=1}^K u_k^\top (\nabla_{D_k} E) v_k + C. \quad (\text{A.13})$$

Algorithm A.1 details the different step used in our algorithm to update  $\{u_k\}_k$  and  $\{v_k\}_k$ .

## A.2 Details on the Z-step optimization

### A.2.1 The coordinate update

**Proposition 1.** *The optimal update  $z'_{k_0}[t_0]$  of the coefficient  $(k_0, t_0)$  is given by*

$$z'_{k_0}[t_0] = \frac{1}{\|D_{k_0}\|_2^2} \max(\beta_{k_0}[t_0] - \lambda, 0),$$

with  $\beta_{k_0}[t_0] = D_{k_0}^\top \tilde{*} \left( X - \sum_{k=1}^K z_k * D_k + z_{k_0}[t_0] e_{t_0} * D_{k_0} \right) [t_0]$  and where  $e_{t_0}$  is the canonical vector in  $\mathbb{R}^{\tilde{T}}$  with value 1 in  $t_0$  and value 0 elsewhere.

*Proof.* For  $y \in \mathbb{R}^+$ , we will denote  $e_{k_0, t_0}(y)$  the cost difference between our current solution estimate  $z_k$  and the signal  $z_k^{(1)}$  where the coefficient  $z_{k_0}[t_0]$  has been replaced by  $y$ , i.e.,

$$z_k^{(1)}[t] = \begin{cases} y, & \text{if } (k, t) = (k_0, t_0) \\ z_k[t], & \text{elsewhere} \end{cases}.$$

Let  $\alpha_{k_0}[t] = (X - \sum_{k=1}^K z_k * D_k)[t] + D_{k_0}[t - t_0] z_{k_0}[t_0]$  for all  $t \in \llbracket 0, T - 1 \rrbracket$ . This quantity denotes the residual when  $z_{k_0}[t_0]$  is set to 0. It is important to note that it can be re-written as,

$$\alpha_k[t] = \left( X - \sum_{k=1}^K z_k * D_k + z_{k_0}[t_0] e_{t_0} * D_{k_0} \right) [t]$$

and thus,  $\beta_{k_0}[t_0] = (D_{k_0}^\dagger \tilde{*} \alpha_{k_0})[t_0]$ . The cost difference  $e_{k_0, t_0}(y)$  is,

$$\begin{aligned}
e_{k_0, t_0}(y) &= \frac{1}{2} \sum_{t=0}^{T-1} \left( X - \sum_{k=1}^K z_k * D_k \right)^2[t] + \lambda \sum_{k=1}^K \|z_k\|_1 - \frac{1}{2} \sum_{t=0}^{T-1} \left( X - \sum_{k=1}^K z_k^{(1)} * D_k \right)^2[t] + \lambda \sum_{k=1}^K \|z_k^{(1)}\|_1 \\
&= \frac{1}{2} \sum_{t=0}^{T-1} (\alpha_{k_0}[t] - D_{k_0}[t - t_0] z_{k_0}[t_0])^2 - \frac{1}{2} \sum_{t=0}^{T-1} (\alpha_{k_0}[t] - D_{k_0}[t - t_0] y)^2 + \lambda(|z_{k_0}[t_0]| - |y|) \\
&= \frac{1}{2} \sum_{t=0}^{T-1} D_{k_0}[t - t_0]^2 (z_{k_0}[t_0]^2 - y^2) - \sum_{t=0}^{T-1} \alpha_{k_0}[t] D_{k_0}[t - t_0] (z_{k_0}[t_0] - y) + \lambda(|z_{k_0}[t_0]| - |y|) \\
&= \frac{\|D_{k_0}\|_2^2}{2} (z_{k_0}[t_0]^2 - y^2) - \underbrace{(D_{k_0}^\dagger \tilde{*} \alpha_{k_0})[t_0]}_{\beta_{k_0}[t_0]} (z_{k_0}[t_0] - y) + \lambda(|z_{k_0}[t_0]| - |y|)
\end{aligned}$$

Using this result, we can derive the optimal value  $z'_{k_0}[t_0]$  to update the coefficient  $(k_0, t_0)$  as the solution of the following optimization problem:

$$z'_{k_0}[t_0] = \arg \max_{y \in \mathbb{R}^+} e_{k_0, t_0}(y) \sim \arg \min_{u \in \mathbb{R}^+} \frac{\|D_{k_0}\|_2^2}{2} \left( y - \frac{\beta_{k_0}[t_0]}{\|D_{k_0}\|_2^2} \right)^2 + \lambda y. \quad (\text{A.14})$$

Simple computations show the desired result, *i.e.*,

$$z'_{k_0}[t_0] = \frac{1}{\|D_{k_0}\|_2^2} \max(\beta_{k_0}[t_0] - \lambda, 0)$$

□

## A.2.2 The $\beta$ update

**Proposition 2.** When updating the coefficient  $z_{k_0}[t_0]$  to the value  $z'_{k_0}[t_0]$ ,  $\beta$  is updated with:

$$\beta_k^{(q+1)}[t] = \beta_k^{(q)}[t] + (D_{k_0}^\dagger \tilde{*} D_k)[t - t_0] (z_{k_0}[t_0] - z'_{k_0}[t_0]), \quad \forall (k, t) \neq (k_0, t_0). \quad (\text{A.15})$$

*Proof.* The value of  $\beta_{k_0}[t_0]$  is independent of the value of  $z_{k_0}[t_0]$ . Indeed, the term  $z_{k_0}[t_0] e_{t_0} * D_{k_0}$  cancel the contribution of  $z_{k_0}[t_0]$  in the convolution  $z_{k_0} * D_{k_0}$ . Thus, when updating the value of the coefficient  $z_{k_0}[t_0]$ ,  $\beta_{k_0}[t_0]$  is not updated.

We denote  $z_k^{(q+1)}$  the activation signal where the coefficient  $z_{k_0}[t_0]$  as been updated to  $z'_{k_0}[t_0]$ , *i.e.*,

$$z_k^{(q+1)}[t] = \begin{cases} z'_{k_0}[t_0], & \text{if } (k, t) = (k_0, t_0) \\ z_k[t], & \text{elsewhere} \end{cases}.$$

For  $(k, t) \neq (k_0, t_0)$ ,

$$\begin{aligned}
\beta_k^{(q+1)}[t] &= \left[ D_k^\dagger \tilde{*} \left( X - \sum_{l=1}^K z_l^{(1)} * D_l + z_k[t] e_t * D_k \right) \right][t] \\
&= \left[ D_k^\dagger \tilde{*} \left( X - \sum_{l=1}^K z_l * D_l + z_k[t] e_t * D_k + (z_{k_0}[t_0] - z'_{k_0}[t_0]) e_{t_0} * D_k \right) \right][t] \\
&= \left[ D_k^\dagger \tilde{*} \left( X - \sum_{l=1}^K z_l * D_l + z_k[t] e_t * D_k \right) \right][t] + \left[ D_k^\dagger \tilde{*} ((z_{k_0}[t_0] - z'_{k_0}[t_0]) e_{t_0} * D_k) \right][t] \\
&= \beta_k^{(q)}[t] + (z_{k_0}[t_0] - z'_{k_0}[t_0]) \left[ D_k^\dagger \tilde{*} (e_{t_0} * D_k) \right][t] \\
&= \beta_k^{(q)}[t] + (D_k^\dagger \tilde{*} D_k)[t - t_0] (z_{k_0}[t_0] - z'_{k_0}[t_0])
\end{aligned}$$

With this relation, it is possible to keep  $\beta_k$  up to date with few operation after each coordinate update. □



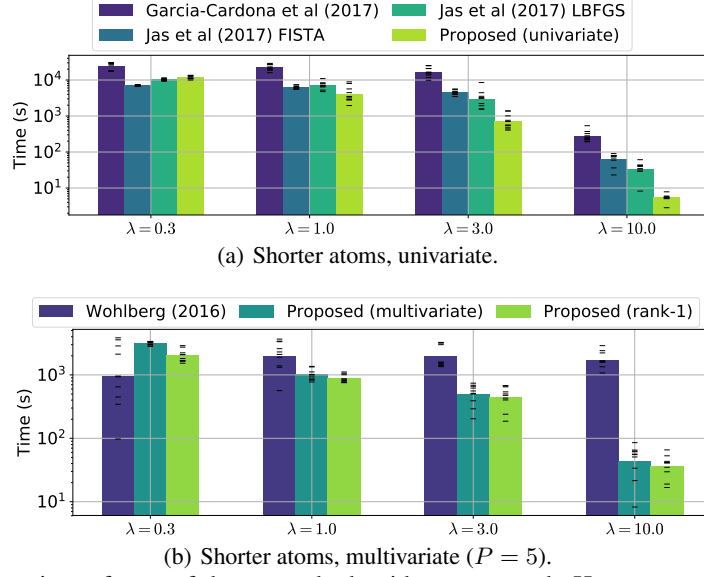


Figure B.1: Comparison of state-of-the-art methods with our approach. Here we used shorter ( $L = 16$  instead of  $L = 128$ ) atoms.

### A.2.3 Precomputation for $D_k^{\dagger} \tilde{*} D_l$

Similarly to the  $D$ -step precomputations, we can precompute  $D_k^{\dagger} \tilde{*} D_l \in \mathbb{R}^{2L-1}$  to speed up the LGCD iterations during the  $Z$ -step. We have:

$$(D_k^{\dagger} \tilde{*} D_l)[t] = \sum_{p=1}^P \sum_{\tau=1}^L D_{k,p}[\tau] D_{l,p}[t + \tau - 1], \quad \forall t \in \llbracket 1, 2L - 1 \rrbracket. \quad (\text{A.16})$$

In the case of the rank-1 constraint model, we can factorize the computation with:

$$(D_k^{\dagger} \tilde{*} D_l)[t] = \left( \sum_{p=1}^P u_{k,p} u_{l,p} \right) \sum_{\tau=1}^L v_k[\tau] v_l[t + \tau - 1], \quad \forall t \in \llbracket 1, 2L - 1 \rrbracket. \quad (\text{A.17})$$

The computational complexities are respectively  $\mathcal{O}(K^2 L^2 P)$  and  $\mathcal{O}(K^2 L(L + P))$ .

## B Additional Experiments

### B.1 Speed performance

We present here more benchmarks as described in [section 4](#), yet with different settings.

First we used shorter atoms of length  $L = 16$  instead of  $L = 128$ , and results are presented in [Figure B.1](#). They confirm the competitiveness of our method, especially when using large regularization parameters. On these problems, the maximum possible regularization  $\lambda_{max}$  was around 90.

Then, we used shorter signals of length  $T = 13\,470$  instead of  $T = 134\,700$  and results are presented in [Figure B.2](#). They also confirm the competitiveness of our method, except with small regularization parameters. However, as the maximum possible regularization  $\lambda_{max}$  was around 90, we question the practical use of these low values, which would poorly enforce the sparsity constraint.

### B.2 Somatosensory dataset

In [Figure 4\(d\)](#), we showed mu-shaped atoms in the *primary* somatosensory region for the MNE somatosensory dataset. Intriguingly, we also find such atoms in the *secondary* somatosensory region, also known as S2. One such atom is shown in [Figure B.3](#).

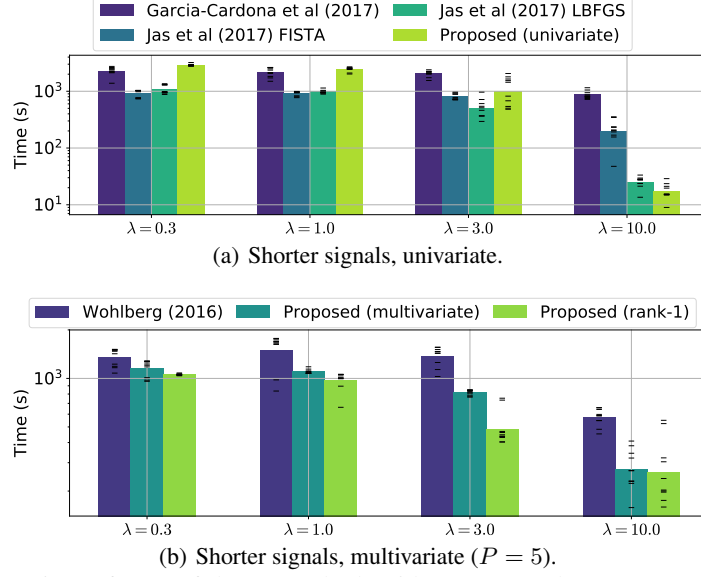


Figure B.2: Comparison of state-of-the-art methods with our approach. Here we used shorter signals ( $T = 13\,470$  instead of  $T = 134\,700$ ).

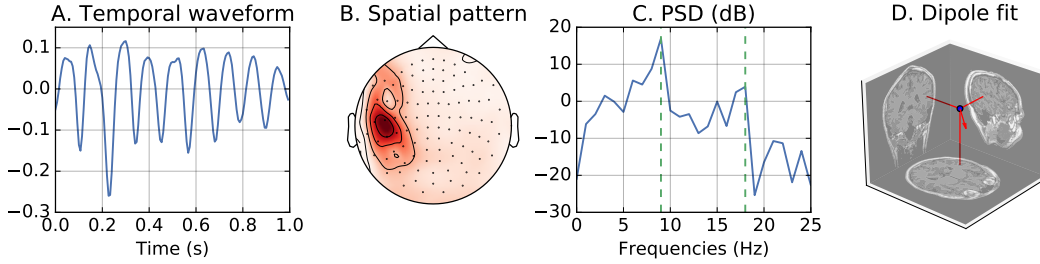


Figure B.3: Atom in the S2 region revealed in the MNE somatosensory data. A. The temporal waveform, and its corresponding B. Spatial pattern, C. The Power Spectral Density (PSD), and D. the dipole fit in the S2 region.

### B.3 Sample dataset

In addition to the MNE somatosensory dataset, we also analyzed the MNE sample dataset [Gramfort et al. \(2013, 2014\)](#). In this case, we used  $N = 1$ , and the number of time points  $T = 41584$  corresponds to 278 s of recording sampled at 150.15 Hz. The magnetometer channels are selected so that the number of channels  $P = 102$ . We learn  $K = 25$  atoms. The sample data is lowpass filtered at 40 Hz, and highpass filtered at 1 Hz.

In [Figure B.6](#), we show the atoms learned on the MNE sample data. [Figure B.6.A](#) shows the temporal waveforms of these atoms and [Figure B.6.C](#) shows the corresponding spatial pattern for a selection of the total atoms. As expected, we are able to recover latent components corresponding to ocular (3rd row) and cardiac artifacts (4th row). Indeed, the ocular artifacts displays the prototypical dipolar pattern in the frontal channels. In [Figure B.6.B](#), we also show the sparse activations associated with the atoms.

More interestingly, we also recover an oscillatory waveform (first row) which appears to originate due to a dipole below the parietal channels at around a frequency of 30 Hz. We confirm this in [Figure B.7](#) using a dipole fit. Indeed, the atom does originate in the parietal lobe which suggests that what we observe is probably a motor rhythm. The dataset under consideration did in fact contain a button press task which could explain the presence of such an atom.

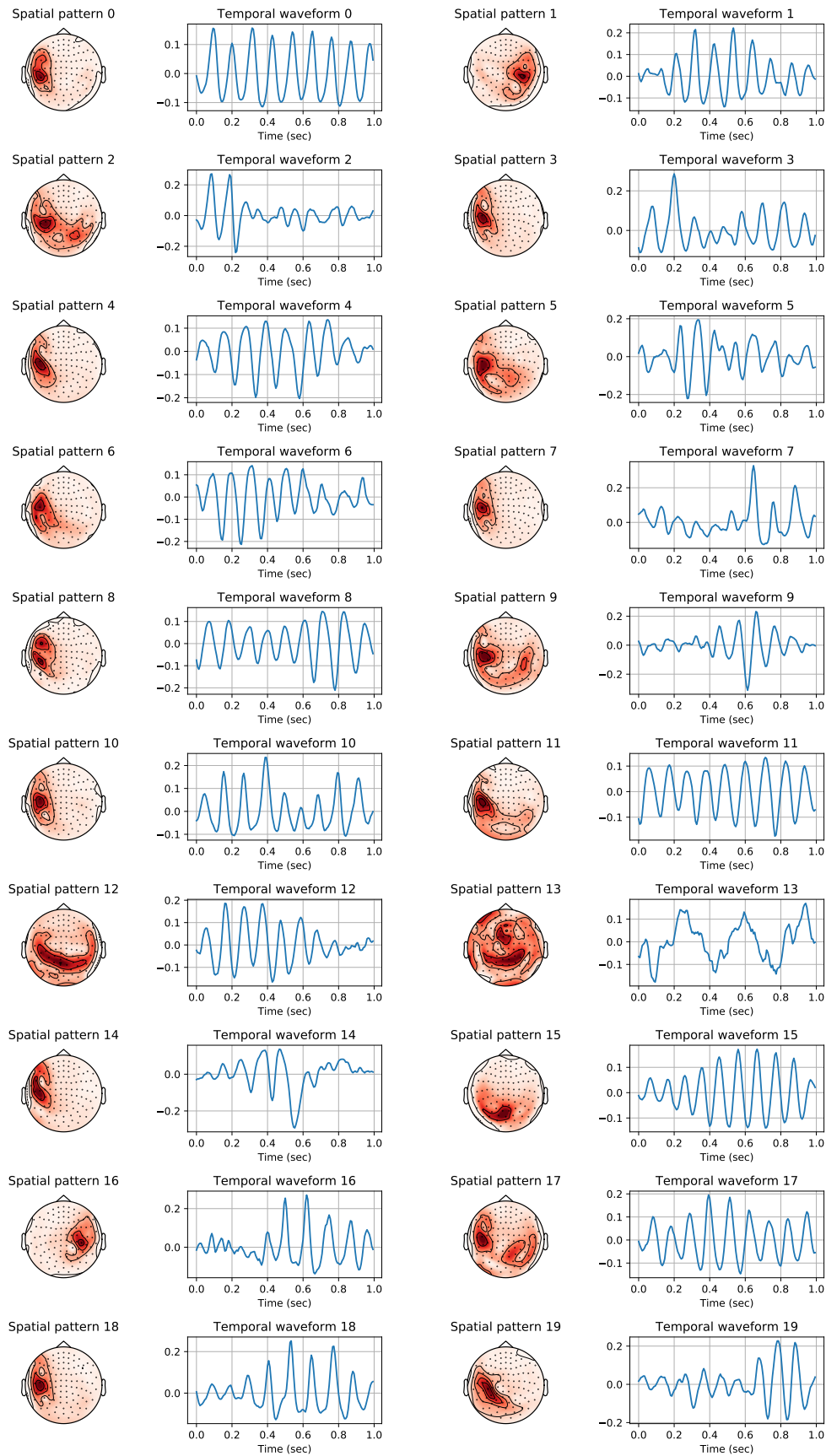


Figure B.4: First 20 atoms learned on the MNE-somatosensory dataset.

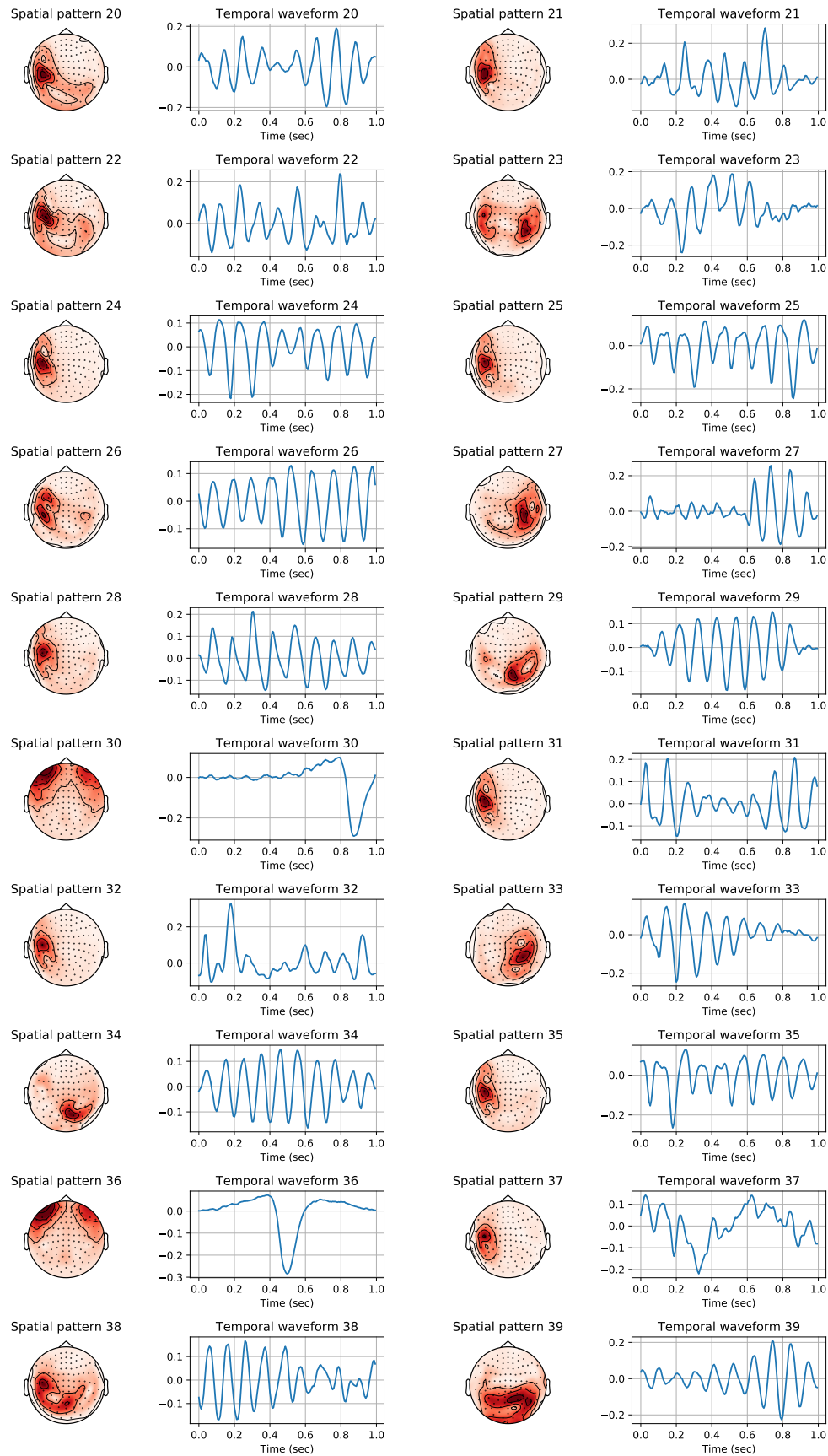


Figure B.5: Last 20 atoms learned on the MNE-somatosensory dataset.

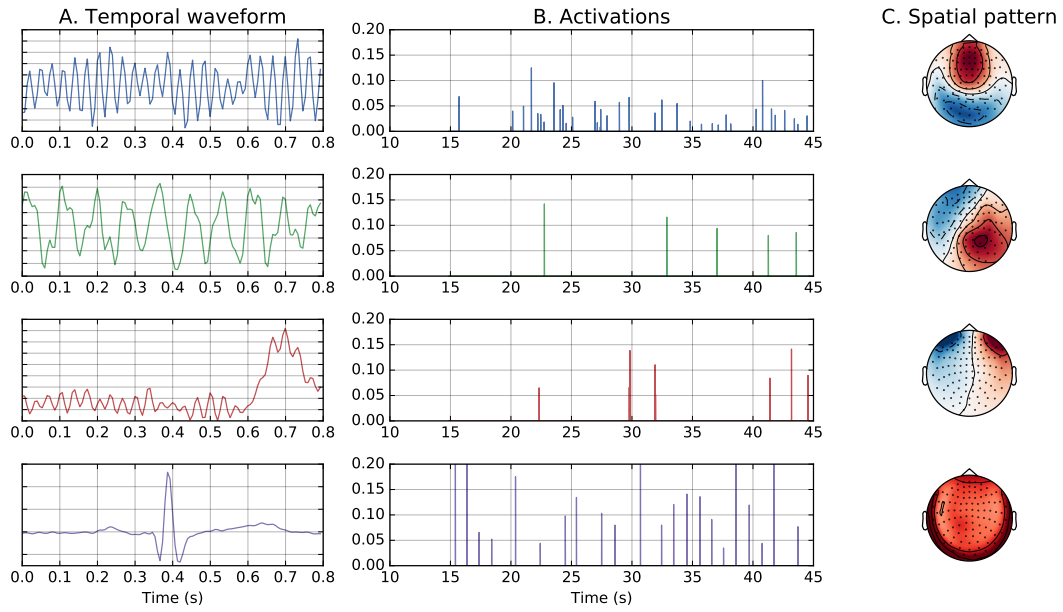


Figure B.6: A selection of A. temporal waveforms of the atoms learned on the MNE sample dataset, and their corresponding B. activations, and C. spatial patterns

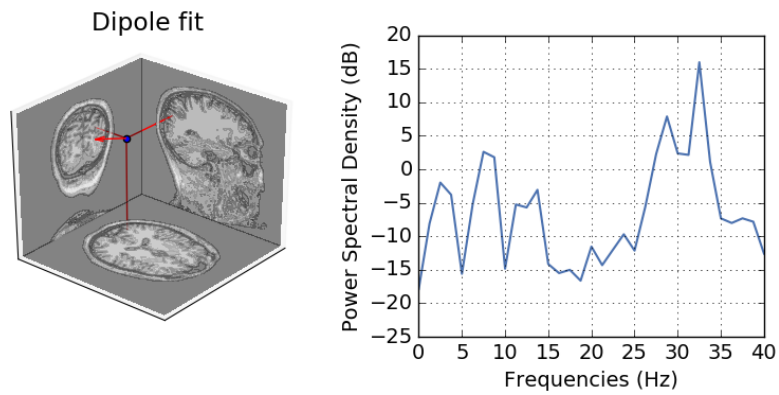


Figure B.7: Dipole fit and power spectral density computed on MNE somato sample dataset for the atom in first row in Figure [Figure B.6](#).