DRIPP: DRIVEN POINT PROCESSES TO MODEL STIMULI INDUCED PATTERNS IN M/EEG SIGNALS

Anonymous authors
Paper under double-blind review

ABSTRACT

The quantitative analysis of non-invasive electrophysiology signals from electroencephalography (EEG) and magnetoencephalography (MEG) boils down to the identification of temporal patterns such as evoked responses, transient bursts of neural oscillations but also blinks or heartbeats for data cleaning. Several works have shown that these patterns can be extracted efficiently in an unsupervised way, *e.g.,* using Convolutional Dictionary Learning. This leads to an event-based description of the data. Given these events, a natural question is to estimate how their occurrences are modulated by certain cognitive tasks and experimental manipulations. To address it, we propose a point process approach. While point processes have been used in neuroscience in the past, in particular for single cell recordings (spike trains), techniques such as Convolutional Dictionary Learning make them amenable to human studies based on EEG/MEG signals. We develop a novel statistical point process model – called driven temporal point processes (DriPP) – where the intensity function of the point process model is linked to a set of point processes corresponding to stimulation events. We derive a fast and principled expectation-maximization (EM) algorithm to estimate the parameters of this model. Simulations reveal that model parameters can be identified from long enough signals. Results on standard MEG datasets demonstrate that our methodology reveals event-related neural responses – both evoked and induced – and isolates non-task specific temporal patterns.

1 INTRODUCTION

Statistical analysis of human neural recordings is at the core of modern neuroscience research. Thanks to non-invasive recording technologies such as electroencephalography (EEG) and magnetoencephalography (MEG), or invasive techniques such as electrocorticography (ECoG) and stereotactic EEG (sEEG), the ambition is to obtain a detailed quantitative description of neural signals at the millisecond timescale when human subjects perform different cognitive tasks (Baillet, 2017). During neuroscience experiments, human subjects are exposed to several external stimuli, and we are interested in knowing how these stimuli influence neural activity.

After pre-processing steps, such as filtering or Independent Component Analysis (ICA; Winkler et al. 2015) to remove artifacts, common techniques relies on epoch averaging – to highlight evoked responses – or time-frequency analysis to quantify power changes in certain frequency bands (Cohen, 2014) – for induced responses. While these approaches has led to numerous neuroscience findings, it has also been criticized. Indeed, averaging tends to blur out the responses due to small jitters in time-locked responses, and the Fourier analysis of different frequency bands tend to neglect the harmonic structure of the signal, leading to the so-called "Fourier fallacy" (Jasper, 1948; Jones, 2016). In so doing, one may conclude to spurious correlation between components that have actually the same origin. Moreover, artifact removal using ICA requires a tedious step of selecting the correct components.

Driven by these drawbacks, a recent trend of work aims to go beyond these classical tools by isolating prototypical waveforms related to the stimuli in the signal (Cole & Voytek, 2017; Dupré la Tour et al., 2018; Donoghue et al., 2020). The core idea consists in decomposing neural signals as combinations of time-invariant patterns, which typically correspond to transient bursts of neural activity (Sherman et al., 2016), or artifacts such as eye blinks or heartbeats. In machine learning, various unsupervised

algorithms have been historically proposed to efficiently identify the patterns and their locations from multivariate temporal signals or images (Lewicki & Sejnowski, 1999; Jost et al., 2006; Heide et al., 2015; Bristow et al., 2013; Wohlberg, 2016b), with applications such as audio classification (Grosse et al., 2007) or image inpainting (Wohlberg, 2016a). For neural signals in particular, several methods have been proposed to tackle this task, such as the sliding window matching (SWM; Gips et al. 2017), the learning of recurrent waveforms (Brockmeier & Príncipe, 2016), adaptive waveform learning (AWL; Hitziger et al. 2017) or convolutional dictionary learning (CDL; Jas et al. 2017; Dupré la Tour et al. 2018). Equipped with such algorithms, the multivariate neural signals are then represented by a set of spatio-temporal patterns, called *atoms*, with their respective onsets, called *activations*. Out of all these methods, CDL has emerged as a convenient and efficient tool to extract patterns, in particular due to its ability to easily include physical priors for the patterns to recover. For example, for M/EEG data, Dupré la Tour et al. (2018) have proposed a CDL method which extracts atoms that appertain to electrical dipole in the brain by imposing a rank-1 structure. While these methods output characteristic patterns and an event-based representation of the temporal dynamic, it is often tedious and requires a certain domain knowledge to quantify how stimuli affect the atoms' activations. Knowing such effects allows to determine whether or not an atom is triggered by a specific type of stimulus, and if so, to quantify by how much, and also with what latency.

As activations are random signals that consist of discrete events, a natural statistical framework is the one of temporal point processes (PP). PP have received a surge of interest in machine learning (Bompaire, 2019; Shchur et al., 2020; Mei et al., 2020) with diverse applications in fields such as healthcare (Lasko, 2014; Lian et al., 2015) or modelling of communities on social networks (Long et al.). In neuroscience, PP have also been studied in the past, in particular to model single cell recordings and neural spike trains (Truccolo et al., 2005; Okatan et al., 2005; Kim et al., 2011; Rad & Paninski, 2011), sometimes coupled with spatial statistics (Pillow et al., 2008) or network models (Galves & Löcherbach, 2015). However, existing models do not directly address our question, namely the characterization of the influence of a deterministic PP – the stimuli onsets – onto a stochastic one – the neural activations derived from M/EEG recordings.

In this paper, we propose a novel method – called driven point process (DriPP) – to model the activation probability for CDL. This method is inspired from Hawkes processes (HP; Hawkes 1971), and models the intensity function of a stochastic process conditioned on the realization of a set of PP, called *drivers*. The resulting process is not auto-excitatory as in HP but can capture the surge of activations associated to external events. In order to model latency effects in neural responses, we propose a parametric model of the intensity function using truncated Gaussian kernels. We derive an efficient expectation-maximization (EM) based inference algorithm and show on synthetic data that it reliably estimates the model parameters. Finally, the evaluation of DriPP on standard MEG datasets shows that it reveals clear neural responses linked to stimuli that can be mapped precisely both in time and in brain space. Our methodology offers a unified approach to extract waveforms and decide if they are unrelated to a cognitive task, such as artifacts or spontaneous brain activity, or if they are provoked by a stimulus – no matter if they are 'evoked' or 'induced' as more commonly described in the neuroscience literature (Tallon-Baudry et al., 1996). While these different effects are commonly extracted using different analysis pipelines, DriPP simply reveals them as stimuli-induced neural responses using a single unified method.

2 Driven temporal point process (DriPP)

A temporal point process (PP) is a stochastic process whose realization consists of discrete events $\{t_i\}$ occurring in continuous time, $t_i \in \mathbb{R}^+$ (Daley & Vere-Jones, 2003). In case where the probability that an event occurs at time t only depends on the past events $\mathscr{F}_t \coloneqq \{t_i,\ t_i < t\}$, PP are usually characterized through the conditional intensity function $\lambda : \mathbb{R}^+ \to \mathbb{R}^+$:

$$\lambda\left(t|\mathscr{F}_t\right) \coloneqq \lim_{\mathrm{d}t \to 0} \frac{\mathbb{P}\left(N_{t+\mathrm{d}t} - N_t = 1|\mathscr{F}_t\right)}{\mathrm{d}t} \ , \tag{1}$$

where $N_t \coloneqq \sum_{i \geq 1} \mathbf{1}_{t_i \leq t}$ is the counting process associated to the PP. This function corresponds to the expected infinitesimal rate at which events are occurring at time t given the arrival times of past events prior to t (Daley & Vere-Jones, 2003).

The proposed model DriPP is adapted from the Hawkes process (HP; Hawkes 1971), as the occurrence of a past event in the driver increases the likelihood of occurrence of activation events in the near

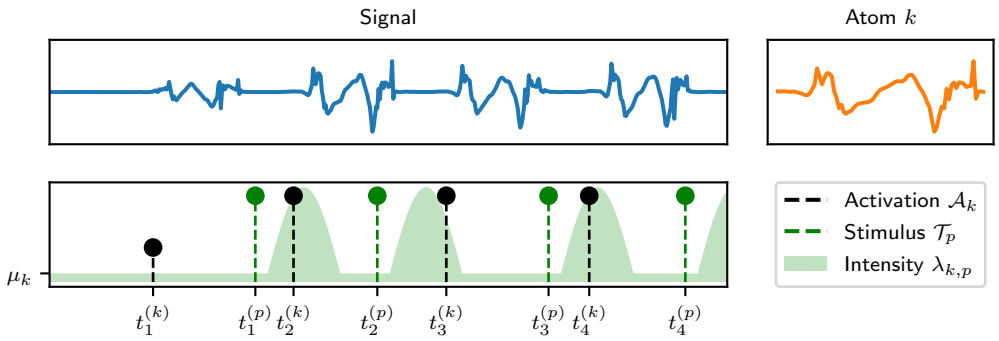

Figure 1: **Top:** CDL applied to a univariate signal (blue) decomposes it as the convolution of a temporal pattern (orange) and a sparse activation signal (black). **Bottom:** Intensity function $\lambda_{k,p}$ defined by its baseline μ_k and the stimulus events $\mathcal{T}_p$ (green). Intensity increases following stimulation events with a certain latency.

future. However, here we suppose that the stochastic point process in our model of neural activations does not have the self-excitatory behavior characteristic of HP. Instead, the sources of activation in the DriPP model are either the drivers or some spontaneous background activity, but not its own previous activations. More specifically, in DriPP, the intensity function at time t between a stochastic process k – whose events are denoted $\mathcal{A}_k = \{t_i^{(k)}\}$ – and a non-empty set of drivers $\mathcal{P}$ – whose events are denoted $\mathcal{T}_p := \{t_1^{(p)}, \dots, t_{n_p}^{(p)}\}, p \in \mathcal{P}$ – is composed of a baseline intensity $\mu_k \geq 0$ and triggering kernels $\kappa_{k,p} : \mathbb{R}^+ \to \mathbb{R}$:

$$\lambda_{k,\mathcal{P}}(t) = \mu_k + \sum_{p \in \mathcal{P}} \sum_{i, t_i^{(p)} \leq t} \alpha_{k,p} \kappa_{k,p} \left(t - t_i^{(p)} \right) \quad , \tag{2}$$

where $\alpha_{k,p} \geq 0$ is a coefficient which controls the relative importance of the driver p on the occurrence of events on the stochastic process k. Note that when the driver processes are known, the intensity function is deterministic, and thus corresponds to the intensity of a inhomogeneous Poisson process (Daley & Vere-Jones, 2003). The coefficient $\alpha_{k,p}$ is set to be non-negative so that we only model excitatory effects, as events on the driver only increase the likelihood of occurrence of new events on the stochastic process. Inhibition effects are assumed non-existent. Figure 1 illustrates how events $\mathcal{T}_p$ on the driver influence the intensity function, after a short latency period.

A critical parametrization of this model is the choice of the triggering kernels $\kappa_{k,p}$. To best model the latency, we use a parametric truncated normal distribution of mean $m_{k,p} \in \mathbb{R}$ and standard deviation $\sigma_{k,p} > 0$, with support $[a\,;b] \subset \mathbb{R}^+$, $b > a$. Namely,

$$\kappa_{k,p}(x) := \kappa\left(x; m_{k,p}, \sigma_{k,p}, a, b\right) = \frac{1}{\sigma_{k,p}} \frac{\phi\left(\frac{x - m_{k,p}}{\sigma_{k,p}}\right)}{\Phi\left(\frac{b - m_{k,p}}{\sigma_{k,p}}\right) - \Phi\left(\frac{a - m_{k,p}}{\sigma_{k,p}}\right)} \mathbf{1}_{a \leq x \leq b} \quad , \tag{3}$$

where ϕ (resp. Φ) is the probability density function (resp. cumulative distribution function) of the standard normal distribution. Note that the truncation values $a, b \in \mathbb{R}^+$ are supposed independent from both the stochastic process and the drivers, hence they are similar for all kernel $p \in \mathcal{P}$. Indeed, in the context of this paper, those truncation values delimit the time interval during which a neuronal response might occur following an external stimulus. In other words, the interval $[a\,;b]$ denotes the range of possible latency values. In the following, we denote by $T := T^{(k)}$ the duration of the process k.

3 Parameters inference with an EM-based algorithm

We propose to infer model parameters $\Theta_{k,\mathcal{P}} = (\mu_k, \boldsymbol{\alpha}_{k,\mathcal{P}}, \boldsymbol{m}_{k,\mathcal{P}}, \boldsymbol{\sigma}_{k,\mathcal{P}})$, where we denote in bold the vector version of the parameter, *i.e.,* $\boldsymbol{x}_{k,\mathcal{P}} = (x_{k,p})_{p \in \mathcal{P}}$, via maximum-likelihood using an EM-based

algorithm (Lewis & Mohler, 2011; Xu et al., 2016). The pseudo-code of the algorithm is presented in Algorithm 1. The expectation-maximization (EM) algorithm (Dempster et al., 1977) is an iterative algorithm that allows to find the maximum likelihood estimates (MLE) of parameters in a probabilistic model when the latter depends on non-observable latent variables. First, from (2), we derive the negative log-likelihood of the model (see details in Appendix A.1):

$$\mathcal{L}_{k,\mathcal{P}}\left(\Theta_{k,\mathcal{P}}\right) = \mu_k T + \sum_{p\in\mathcal{P}} \alpha_{k,p} n_p - \sum_{t\in\mathcal{A}_k} \log\left(\mu_k + \sum_{p\in\mathcal{P}} \sum_{i,t_i^{(p)}\leq t} \alpha_{k,p}\kappa_{k,p}\left(t - t_i^{(p)}\right)\right) \quad . \tag{4}$$

Expectation step From a given estimate, the E-step aims at computing the events assignation, *i.e.,* the probability that an event comes from either the kernel or the baseline intensity. At iteration n, let $P_k^{(n)}(t) \equiv P_k^{(n)}(t;p,k)$ be the probability that the activation at time $t \in [0\,;T]$ has been triggered by the baseline intensity of stochastic process k, and $P_p^{(n)}(t) \equiv P_p^{(n)}(t;p,k)$ be the probability that the activation at time t has been triggered by the driver p. By definition of our intensity model (2), we have:

$$P_k^{(n)}(t) = \frac{\mu_k^{(n)}}{\lambda_{k,\mathcal{P}}^{(n)}(t)} \qquad \text{and} \qquad \forall p \in \mathcal{P}, P_p^{(n)}(t) = \frac{\alpha_{k,p}^{(n)} \sum_{i,t_i^{(p)}\leq t} \kappa_{k,p}^{(n)}\left(t - t_i^{(p)}(t)\right)}{\lambda_{k,\mathcal{P}}^{(n)}(t)} \quad , \tag{5}$$

where $\theta^{(n)}$ denotes the value of the parameter θ at step n of the algorithm, and similarly, if f is a function of parameter θ, $f^{(n)}\left(x;\theta\right) \coloneqq f\left(x;\theta^{(n)}\right)$. Note that $\forall t \in [0\,;T]\,, P_k^{(n)}(t) + \sum_{p\in\mathcal{P}} P_p^{(n)}(t) = 1$.

Maximization step Once this assignation have been computed, one needs to update the parameter of the model using MLE. To obtain the update equations, we fix the probabilities $P_k^{(n)}$ and $P_p^{(n)}$, and cancel the negative log-likelihood derivatives with respect to each parameter. For given value of probabilities $P_k^{(n)}(t)$ and $P_p^{(n)}(t)$, we here derive succinctly the update for parameters μ and $\boldsymbol{\alpha}$:

$$\frac{\partial \mathcal{L}_{k,\mathcal{P}}}{\partial \mu_k^{(n)}}\left(\Theta_{k,\mathcal{P}}^{(n)}\right) = 0 \Leftrightarrow T - \sum_{t\in\mathcal{A}_k} \frac{1}{\lambda_{k,\mathcal{P}}^{(n)}(t)} = 0 \Leftrightarrow T - \sum_{t\in\mathcal{A}_k} \frac{P_k^{(n)}(t)}{\mu_k^{(n)}(t)} = 0 \Leftrightarrow \mu_k^{(n+1)} = \frac{1}{T}\sum_{t\in\mathcal{A}_k} P_k^{(n)}(t) \tag{6}$$

$$\frac{\partial \mathcal{L}_{k,\mathcal{P}}}{\partial \alpha_{k,p}^{(n)}}\left(\Theta_{k,\mathcal{P}}^{(n)}\right) = 0 \Leftrightarrow n_p - \sum_{t\in\mathcal{A}_k} \frac{P_p^{(n)}(t)}{\lambda_{k,\mathcal{P}}^{(n)}(t)} = 0 \Leftrightarrow \alpha_{k,p}^{(n+1)} = \pi_{\mathbb{R}^+}\left(\frac{1}{n_p}\sum_{t\in\mathcal{A}_k} P_p^{(n)}(t)\right) \tag{7}$$

These two updates amount to maximizing the probabilities that the events assigned to the driver or the baseline stay assigned to the same generation process. Then, we give the update equations for $\boldsymbol{m}$ and $\boldsymbol{\sigma}$, which corresponds to parametric estimates of each truncated Gaussian kernel parameters with events assigned to the kernel. Detailed computations are provided in Appendix A.2.

$$m_{k,p}^{(n+1)} = \frac{\sum_{t\in\mathcal{A}_k}\sum_{i,t_i^{(p)}\leq t} \frac{\alpha_{k,p}^{(n)}\left(t-t_i^{(p)}\right)\kappa_{k,p}^{(n)}\left(t-t_i^{(p)}\right)}{\lambda_{k,\mathcal{P}}^{(n)}(t)}}{\sum_{t\in\mathcal{A}_k} P_p^{(n)}(t)} - {\sigma_{k,p}^{(n)}}^2 \frac{C_m\left(m_{k,p}^{(n)}, \sigma_{k,p}^{(n)}, a, b\right)}{C\left(m_{k,p}^{(n)}, \sigma_{k,p}^{(n)}, a, b\right)} \tag{8}$$

$$\sigma_{k,p}^{(n+1)} = \pi_{[\varepsilon\,;+\infty)}\left(\frac{C\left(m_{k,p}^{(n)}, \sigma_{k,p}^{(n)}, a, b\right)}{C_\sigma\left(m_{k,p}^{(n)}, \sigma_{k,p}^{(n)}, a, b\right)} \frac{\sum_{t\in\mathcal{A}_k}\sum_{i,t_i^{(p)}\leq t} \frac{\alpha_{k,p}^{(n)}\left(t-t_i^{(p)}(t)-m_{k,p}^{(n)}\right)^2\kappa_{k,p}^{(n)}\left(t-t_i^{(p)}\right)}{\lambda_{k,\mathcal{P}}^{(n)}(t)}}{\sum_{t\in\mathcal{A}_k} P_p^{(n)}(t)}\right)^{1/3} \tag{9}$$

where

$$C\left(m,\sigma,a,b\right) \coloneqq \int_a^b \exp\left(-\frac{1}{2}\frac{(u-m)^2}{\sigma^2}\right)\mathrm{d}u \quad , \tag{10}$$

and $C_m\left(m,\sigma,a,b\right) \coloneqq \frac{\partial C}{\partial m}\left(m,\sigma,a,b\right)$ and $C_\sigma\left(m,\sigma,a,b\right) \coloneqq \frac{\partial C}{\partial \sigma}\left(m,\sigma,a,b\right)$. Here, $\pi_E(\cdot)$ denotes the projection operator on the ensemble E, and $\varepsilon > 0$ is predetermined to ensure that σ remains strictly positive. In practice, we set ε such that we avoid the overfitting that can occur when the kernel's mass is too concentrated. Note that once the initial values of the parameters are determined, the EM algorithm is entirely deterministic.

Algorithm 1: EM-based algorithm

input : $\mathcal{A}_k, \mathcal{T}_p, a, b, T, N$
output : The estimated values for parameters $\mu, \boldsymbol{\alpha}, \boldsymbol{m}$ and $\boldsymbol{\sigma}$

Initialize $\mu^{(0)}, \boldsymbol{\alpha}^{(0)}, \boldsymbol{m}^{(0)}, \boldsymbol{\sigma}^{(0)}$;
for $i = 0, \dots, N-1$ **do**
  if $\boldsymbol{\alpha}^{(i)} = \mathbf{0}_{\mathbb{R}^{\#\mathcal{P}}}$ **then**
    $\mu^{(i+1)} = \mu^{(\mathrm{MLE})}$ using Equation (11);
    break;
  end
  Define $\lambda^{(i)}$ using Equation (2);
  Compute $\mu^{(i+1)}, \boldsymbol{\alpha}^{(i+1)}, \boldsymbol{m}^{(i+1)}, \boldsymbol{\sigma}^{(i+1)}$;
end
return $\mu^{(i+1)}, \boldsymbol{\alpha}^{(i+1)}, \boldsymbol{m}^{(i+1)}, \boldsymbol{\sigma}^{(i+1)}$

Also, when the estimate of parameter m is too far from the kernel's support $[a\,;b]$, we are in a pathological case where EM is diverging due to indeterminacy between setting $\alpha = 0$ and pushing m to infinity due to the discrete nature of our events. Thus, we consider that the stochastic process is not linked to the considered driver, and fall back to the MLE estimator. The algorithm is therefore stopped and we set $\boldsymbol{\alpha} = \mathbf{0}_{\mathbb{R}^{\#\mathcal{P}}}$.

It is worth noting that if $\forall p \in \mathcal{P}, \alpha_{k,p} = 0$, then the intensity is reduced to its baseline, thus the negative log-likelihood is $\mathcal{L}_{k,p}(\Theta_{k,p}) = \mu_k T - \#\mathcal{A}_k \log \mu_k$, where $\#\mathcal{A}$ denotes the cardinality of the set $\mathcal{A}$. Thus, we can complete the EM algorithm by directly computing the MLE for μ_k, namely:

$$\mu_k^{(\mathrm{MLE})} = \frac{\#\mathcal{A}_k}{T} \quad . \tag{11}$$

Initialization strategy

We propose a "smart start" initialization strategy, where parameters are initialized based on their role in the model. It reads:

$$\mu_k^{(0)} = \frac{\#\mathcal{A}_k - \#\left(\bigcup_{p\in\mathcal{P}} \mathcal{D}_{k,p}\right)}{T - \boldsymbol{\lambda}\left(\bigcup_{p\in\mathcal{P}} \bigcup_{t'\in\mathcal{T}_p} [t'+a\,;t'+b]\right)} \tag{12}$$

$$\alpha_{k,p}^{(0)} = \frac{\#\mathcal{D}_{k,p}}{\boldsymbol{\lambda}\left(\bigcup_{t'\in\mathcal{T}_p} [t'+a\,;t'+b]\right)} - \mu_k^{(0)}, \quad \forall p \in \mathcal{P} \tag{13}$$

$$m_{k,p}^{(0)} = \frac{1}{\#\mathcal{D}_{k,p}} \sum_{d\in\mathcal{D}_{k,p}} d \quad \text{and} \quad \sigma_{k,p}^{(0)} = \sqrt{\frac{1}{\#\mathcal{D}_{k,p}} \sum_{d\in\mathcal{D}_{k,p}} \left|d - m^{(0)}\right|^2}, \quad \forall p \in \mathcal{P} \quad , \tag{14}$$

where $\boldsymbol{\lambda}(\cdot)$ denotes the Lebesgue measure, and where $\mathcal{D}_{k,p} := \{t - t_*^{(p)}(t),\ t \in \mathcal{A}_k\} \cap [a\,;b]$ is the set of all empirical delays possibly linked to the driver p, with $t_*^{(p)}(t) := \max\{t',\ t' \in \mathcal{T}_p, t' \leq t\}$ denoting the timestamp of the last event on the driver p that occurred before time t. Here, the baseline intensity $\mu^{(0)}$ is set to the average number of process' events that occur outside any kernel support, *i.e.,* the events that are guaranteed to be exogenous or spontaneous. Similarly, the kernel intensity $\alpha^{(0)}$ is computed as the increase in the average number of activations over the kernel support, compared to $\mu^{(0)}$. The initial guess for $m^{(0)}$ and $\sigma^{(0)}$ are obtained with their parametric estimate considering that all event on the kernel support are assigned to the considered driver.

4 Experiments

We evaluated our model on several experiments, using both synthetic and empirical MEG data. We used `Python` (Python Software Foundation, 2019) and its scientific libraries (Virtanen et al., 2020; Hunter, 2007; Harris et al., 2020). We relied on `alphacsc` for CDL with rank-1 constraints on MEG (Dupré la Tour et al., 2018) and we used `MNE` (Gramfort et al., 2013) to load and manipulate the MEG datasets. Computations were run on CPU Intel(R) Xeon(R) E5-2699, with 44 physical cores.

4.1 Evaluation of the EM convergence on synthetic data

For given number of drivers and a set of corresponding parameters Θ, we first generate the drivers' processes and then simulate the stochastic process for a pre-determined duration T. Each driver's

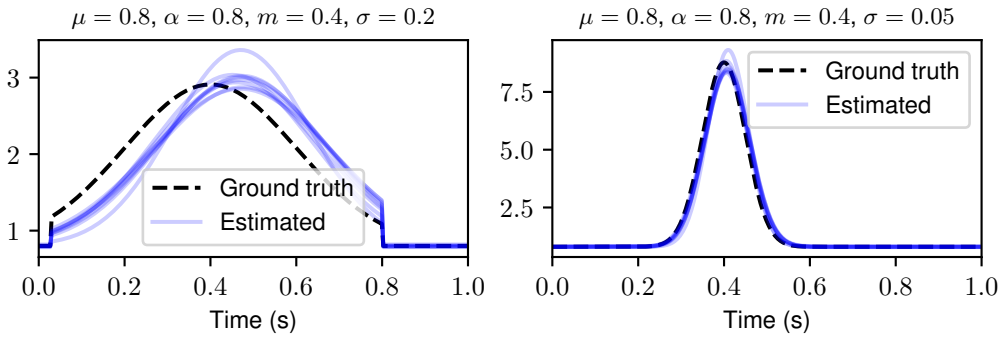

Figure 2: True and estimated intensity functions following a driving event at time zero for two different kernels, on synthetic data. **Left**: "wide" kernel with $\sigma = 0.2$. **Right**: "sharp" kernel with $\sigma = 0.05$. Parameters used are $T = 10000$, $P/S = 0.3$. On synthetic data, the EM algorithm satisfactorily retrieves the true values of parameters, especially for sharp kernels.

timestamps are simulated as follows: given an inter-stimuli interval (ISI), a set of $S = \lfloor \frac{T}{\text{ISI}} \rfloor$ equidistant timestamps is generated – where $\lfloor \cdot \rfloor$ denotes the floor function. Then P timestamps are uniformly sampled without replacement from this set. In all our experiments, we fixed the ISI to $1\,\text{s}$ for the "wide" kernel, and to $1.4\,\text{s}$ for the "sharp" one. Finally, a one-dimensional non-homogeneous Poisson process is simulated following Lewis' thinning algorithm (Lewis & Shedler, 1979), given the predefined intensity function λ and the drivers' timestamps.

Figure 2 illustrates the intensity function recovery with two drivers considered together: the first one has a "wide" kernel with standard deviation $\sigma = 0.2\,\text{s}$, and the second one has a "sharp" kernel with $\sigma = 0.05\,\text{s}$. Both kernels have support $[0.03\,\text{s}\,; 0.8\,\text{s}]$ and mean $m = 0.4\,\text{s}$, the coefficients α are both set to 0.8 and the baseline intensity parameter μ to 0.8. We report 8 estimated intensities obtained from independent simulation of the processes – using $T = 10000\,\text{s}$ and $P/S = 0.3$ – that we plot over each one of the driver's kernel's support. The EM algorithm is run for 50 iterations using the "smart start" initialization strategy described in Section 2. Note that here, the randomness only comes from the data generation, as the EM algorithm uses a deterministic initialization. The EM algorithm is able to recover the parameters up to a satisfactory level, especially for sharp kernels for which the bias appears minimal compared to wide kernels.

In order to provide a quantitative evaluation of the parameters recovery, we compute, for each driver $p \in \mathcal{P}$, the ℓ_∞ norm between the true parameters Θ_p^* and its estimation $\widehat{\Theta}_p$:

$$\left\| \Theta_p^* - \widehat{\Theta}_p \right\|_\infty := \max_{t \in [0\,;T]} \left| \mu^* + \alpha_p^* \kappa_p^*(t) - \hat{\mu} - \hat{\alpha_p} \hat{\kappa}_p(t) \right| \quad . \tag{15}$$

The rationale for using the ℓ_∞ norm is to ensure that errors during baseline and within the kernel support are given equal importance. Figure 3 presents the parameter recovery for the same scenario with varying P/S and T. In order to easily compare the EM performances on the two shapes of kernels, Figure 3 reports the mean of the relative ℓ_∞ norm – that is the ℓ_∞ divided by the maximum of the true intensity λ^* – computed for each of the driver over 30 repetitions with different realization of the process. The results show that the higher the density of events (increase in P/S), the better the parameter estimates are. However, for both shapes of kernel, an increase in the number of data following an increase in the process duration T has a limited impact on the performance. Indeed, the relative infinite norm decreases for the first values of T but rapidly reaches a plateau. The decrease is particularly pronounced for limited event density ($P/S = 0.1$). In contrast, note that the variance of the estimate continually decreases with the size of the simulated process, as reported in Figure A.1 in appendix. Thus, the EM converges to a certain limit that stays at a specific distance from the ground truth. This suggests that the estimators are slightly biased.

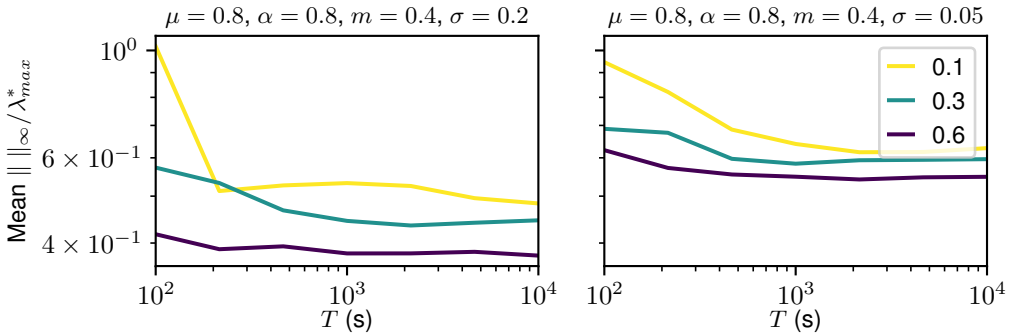

Figure 3: Mean of the relative infinite norm as a function of process duration T and the percentage of events kept P/S, for two kernel shapes on synthetic data: wide kernel (**left**) and sharp kernel (**right**). The EM performance increases with denser processes but reaches a plateau with the increasing duration T.

4.2 Evoked and Induced effects characterization in MEG data

Datasets Experiments on MEG data were run on two datasets from `MNE` Python package (Gramfort et al., 2014; 2013): the *sample* dataset and the somatosensory (*somato*) dataset.[1] These datasets were selected as they elicit two distinct types of event-related neural activations: evoked responses which are time locked to the onset of the driver process, and induced responses which exhibit random jitters. Complementary experiments were performed on the larger Cam-CAN dataset (Shafto et al., 2014)[2]. Presentation of the dataset, data pre-processing and obtained results on 3 subjects are presented in Appendix A.6. The presented results are self determined as they exhibit, for each subject, the atoms that have the higher ratio α/μ. For all studied datasets, full results are presented in supplementary materials.

The *sample* dataset contains M/EEG recordings of a human subject presented with audio and visual stimuli. In this experiment, checker-board patterns are presented to the subject in the left and right visual field, interspersed by tones to the left or right ear. The experiment lasts about $4.6\,\mathrm{min}$ and approximately 70 stimuli per type are presented to the subject. The interval between the stimuli is on average of $750\,\mathrm{ms}$, all types combined, with a minimum of $593\,\mathrm{ms}$. Occasionally a smiley face is presented at the center of the visual field. The subject was asked to press a button with the right index finger as soon as possible after the appearance of the face. In the following, we are only interested in the four main stimuli types: auditory left, auditory right, visual left and visual right.

For the *somato* dataset, a human subject is scanned with MEG during $15\,\mathrm{min}$, while 111 stimulations of his left median nerve were made. The minimum ISI is $7\,\mathrm{s}$.

Experimental setting For both datasets, only the 204 gradiometer channels are analyzed. The signals are pre-processed using high-pass filtering at 2 Hz to remove slow drifts in the data, and are resampled to 150 Hz to limit the atom size in the CDL. CDL is computed using `alphacsc` (Dupré la Tour et al., 2018) with the `GreedyCDL` method. For the *sample* dataset, 40 atoms of duration $1\,\mathrm{s}$ each are extracted, and for the *somato* dataset 20 atoms of duration $0.53\,\mathrm{s}$ are estimated. The extracted atoms' activations are binarized using a threshold of 6×10^{-11} (resp. 1×10^{-10}) for *sample* (resp. *somato*), and the events time are shifted to make them correspond to the peak amplitude time in the atom. Then, for every atom, the intensity function is estimated using the EM-based algorithm with 400 iterations and the "smart start" initialization strategy. Kernels' truncation values are hyper-parameters for the EM and thus must be pre-determined. The upper truncation value b is chosen smaller than the minimum ISI. Here, we used in addition some previous domain knowledge in order to set coherent values for each dataset. Hence, for the *sample* (resp. *somato*) dataset, kernel support is fixed at $[0.03\,\mathrm{s}\,; 0.5\,\mathrm{s}]$ (resp. $[0\,\mathrm{s}\,; 2\,\mathrm{s}]$).

[1] Both available at `https://mne.tools/stable/overview/datasets_index.html`
[2] Available at `https://www.cam-can.org/index.php?content=dataset`

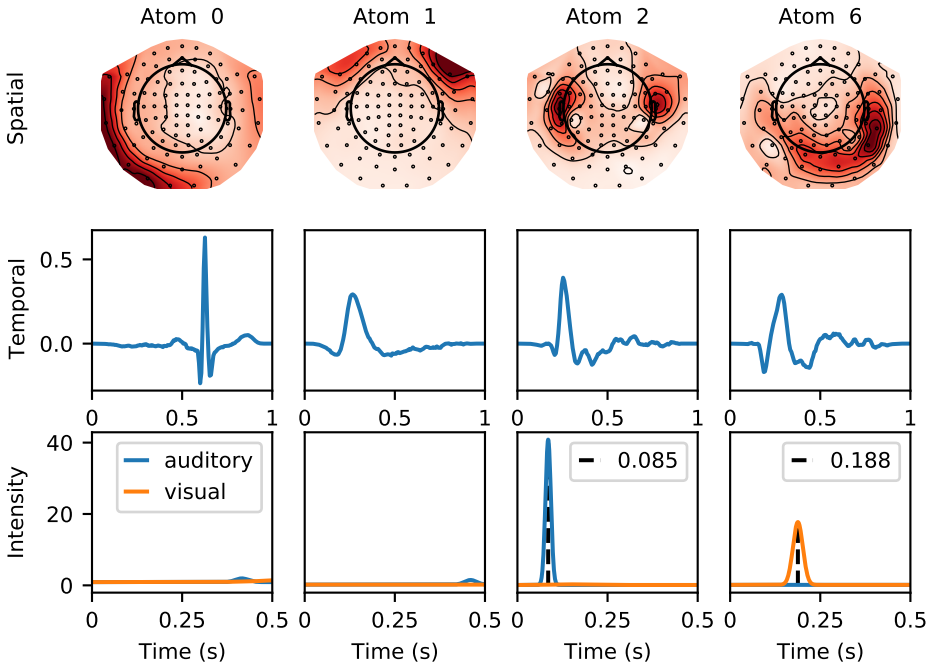

Figure 4: Spatial and temporal patterns of 4 atoms from *sample* dataset, and their respective estimated intensity functions following a stimulus (cue at time = 0 s), for auditory and visual stimuli. The heartbeat and eye-blink artifacts are not linked to any stimuli. An auditory stimulus will induced a neural response similar to atom 2, with a mean latency of $85\,\mathrm{ms}$.

Evoked responses in sample dataset Results on the *sample* dataset are presented in Figure 4. We plot the spatial and temporal representations of 4 selected atoms, as well as the estimated intensity functions related to the two types of stimuli: auditory (*blue*) and visual (*orange*). The first two atoms are specifically handpicked to exhibit usual artifacts, and the last two are selected as they have the two bigger ratios α/μ for their respective learned intensity. Even though the intensity is learned with the two stimuli conjointly, we plot the two corresponding "intensities at kernel" separately, *i.e.,* $\forall p \in \mathcal{P}, \forall t \in [0\,;0.5]$, we plot $\lambda_{k,p}(t), k = 0, 1, 2, 6$.

Spatial and temporal representations of atom 0 (resp. atom 1) indicate that it corresponds to the heartbeat (resp. the eye blinks) artifact. These two atoms are thus expected not to be linked to any stimuli. This is confirmed by the shape of the intensities estimated with DriPP that is mostly flat, which indicates that the activation of these two atoms are independent on auditory and visual stimuli. Note that these two artifacts can also be recover by an Independent Component Analysis (ICA), as shown in Figure A.3. Indeed, the cosine similarity between the eye-blink (resp. the heartbeat) artifact extracted with CDL and its corresponding component in ICA analysis is $98.5\,\%$ (resp. $99.1\,\%$).

In contrast, by looking at the spatial and temporal patterns of atom 2 (resp. atom 6), it can be associated with an auditory (resp. visual) evoked response. Given the spatial topographies of atom 2 we conclude of a bilateral auditory response and the peak transient temporal pattern suggests an evoked response that is confirmed by the estimated intensity function that contains a narrow peak around $85\,\mathrm{ms}$ post-stimulus. This is the N100 response well known in the MEG/EEG literature (Näätänen & Picton, 1987). Regarding atom 6, topography is right lateralized in the occipital region suggesting a visual evoked reponse. This is confirmed by the intensity function estimated that reports a relationship between this atoms and the visual stimuli. Here also the intensity peak is narrow which is characteristic of an evoked response. Moreover, the intensity estimated with DriPP for the unrelated tasks are completely flat. We have $\alpha = 0$, which indicates that atoms' activations are exogenous or spontaneous relatively to the unrelated stimuli.

Induced response in somato dataset Results on the *somato* dataset are presented in Figure 5. Similar to the results on *sample*, spatial and temporal pattern of 3 handpicked atoms are plotted,

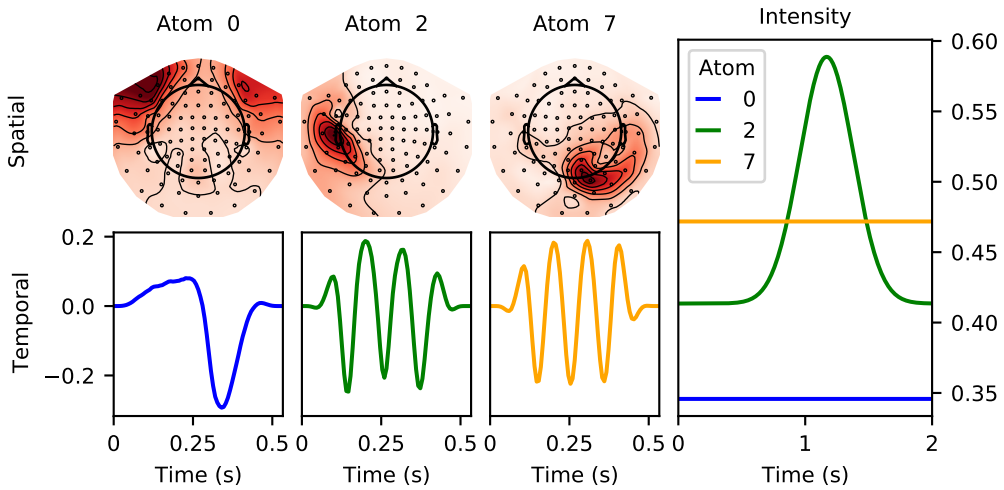

Figure 5: Spatial and temporal patterns of 3 atoms from *somato* dataset, and their respective estimated intensity functions following a somatosensory stimulus (cue at time = 0 s). The eye-blink artifact are not linked to the stimulus. An somatosensory stimulus will induced a neural response similar to atom 2, with a mean latency of 1 s.

alongside the intensity functions obtained with DriPP. Thanks to their spatial and temporal patterns, and with some domain knowledge, it is possible to categorize these 3 atoms: atom 2 corresponds to a μ-wave located in the secondary somatosensory region (S2), atom 7 corresponds to an α-wave originating in the occipital visual areas, whereas atom 0 corresponds to the eye-blink artifact.

As α-waves are spontaneous brain activity, they are not phase-locked with the stimuli. It is thus expected that atom 7 is not linked to the task, as confirmed by its estimated intensity function where $\alpha = 0$. For atom 2 – that corresponds to a μ-wave –, its respective intensity is non-flat with a wide peak close to 1 s, which characterizes an induced response. Moreover, similar to results on the *sample* dataset, we recover the eye-blink artifacts, that also has a flat intensity function. This allows us to be confident in the interpretation of the obtained results. Some other μ-wave atoms – atoms 1 and 4 – are presented in Figure A.7 in Appendix A.5. They have an estimated intensity similar to atom 2, *i.e.,* non-flat with a wide peak close to 1 s. The usual time/frequency analysis reported in Figure A.6 exhibits the induced response of the μ-wave.

5 Discussion

We proposed a point process (PP) based approach specially designed to model how external stimuli can influence the occurrences of recurring spatio-temporal patterns, called *atoms*, extracted from M/EEG recordings using convolutional dictionary learning (CDL). The parametrization of the intensity kernel as a truncated Gaussian allows to model a certain latency between a stimulus and its neural response, which differs from the usual parametrizations in PP models that captures immediate response, *e.g.,* with exponential kernels. Truncation values of the kernel are hyperparameters as they are not learned by the algorithm, and thus must be pre-determined. In the paper we used some domain knowledge to set those values. Exploring methods to infer these hyperparameters based on the data would be an interesting future work. One potential limitation of our work is that we only model potential excitation by setting $\alpha \geq 0$. A future extension of our work would be to allow inhibition effect, but it implies to modify the definition of the intensity function as we have to ensure it stays non-negative at all times, which could potentially degrade the interpretability of the model parameters. Another extension would be to also account for conditional dependencies between the different atoms themselves. This would involve PP closer to multivariate Hawkes processes, yet this would increase the number of parameters which could lead to overfitting given that M/EEG usually last only a few minutes. A final thought of work would be to integrate a part of randomness in the parameters estimation, to better account for the non-convexity of the problem and thus explore other local minima.

REFERENCES

S. Baillet. Magnetoencephalography for brain electrophysiology and imaging. *Nature Neuroscience*, 20:327 EP –, 02 2017.

M. Bompaire. *Machine learning based on Hawkes processes and stochastic optimization*. Theses, Université Paris Saclay (COmUE), July 2019.

H. Bristow, A. Eriksson, and S. Lucey. Fast convolutional sparse coding. In *Computer Vision and Pattern Recognition (CVPR)*, pp. 391–398, 2013.

A. J. Brockmeier and J. C. Príncipe. Learning recurrent waveforms within EEGs. *IEEE Transactions on Biomedical Engineering*, 63(1):43–54, 2016.

M. X. Cohen. *Analyzing Neural Time Series Data: Theory and Practice*. The MIT Press, 01 2014.

S. R. Cole and B. Voytek. Brain Oscillations and the Importance of Waveform Shape. *Trends in Cognitive Sciences*, 21(2):137–149, 2017.

D. J. Daley and D. Vere-Jones. *An introduction to the theory of point processes. Volume I: Elementary theory and methods*. Probability and Its Applications. Springer-Verlag New York, 2003.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

T. Donoghue, M. Haller, E. J. Peterson, P. Varma, P. Sebastian, R. Gao, T. Noto, A. H. Lara, J. D. Wallis, R. T. Knight, A. Shestyuk, and B. Voytek. Parameterizing neural power spectra into periodic and aperiodic components. *Nature Neuroscience*, 23(12):1655–1665, 2020.

T. Dupré la Tour, T. Moreau, M. Jas, and A. Gramfort. Multivariate convolutional sparse coding for electromagnetic brain signals. *Advances in Neural Information Processing Systems*, 31:3292–3302, 2018.

A. Galves and E. Löcherbach. Modeling networks of spiking neurons as interacting processes with memory of variable length. *arXiv preprint arXiv:1502.06446*, 2015.

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, et al. Meg and eeg data analysis with mne-python. *Frontiers in neuroscience*, 7:267, 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)*, pp. 149–158. AUAI Press, 2007. ISBN 0-9749039-3-0.

C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020.

A. G. Hawkes. Point spectra of some mutually exciting point processes. *Journal of the Royal Statistical Society: Series B (Methodological)*, 33(3):438–443, 1971.

F. Heide, W. Heidrich, and G. Wetzstein. Fast and flexible convolutional sparse coding. In *Computer Vision and Pattern Recognition (CVPR)*, pp. 5135–5143. IEEE, 2015.

S. Hitziger, M. Clerc, S. Saillet, C. Bénar, and Papadopoulo T. Adaptive waveform learning: A framework for modeling variability in neurophysiological signals. *IEEE Transactions on Signal Processing*, 65(16):4324–4338, 2017.

J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science Engineering*, 9(3):90–95, 2007.

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)*, pp. 1–15, 2017.

H. H. Jasper. Charting the sea of brain waves. *Science*, 108(2805):343–347, 1948.

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. Vandergheynst, 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.

S. Kim, D. Putrino, S. Ghosh, and E. N. Brown. A granger causality measure for point process models of ensemble neural spiking activity. *PLoS Comput Biol*, 7(3):e1001110, 2011.

T. A. Lasko. Efficient inference of gaussian-process-modulated renewal processes with application to medical event data. In *Uncertainty in artificial intelligence: proceedings of the... conference. Conference on Uncertainty in Artificial Intelligence*, volume 2014, pp. 469. NIH Public Access, 2014.

M. S. Lewicki and T. J. Sejnowski. Coding time-varying signals using sparse, shift-invariant representations. In M. J. Kearns, S. A. Solla, and D. A. Cohn (eds.), *Advances in Neural Information Processing Systems (NIPS)*, pp. 730–736. MIT Press, 1999.

PA W Lewis and Gerald S Shedler. Simulation of nonhomogeneous poisson processes by thinning. *Naval research logistics quarterly*, 26(3):403–413, 1979.

R. Lewis and G. Mohler. A nonparametric em algorithm for multiscale hawkes processes. *Journal of Nonparametric Statistics*, 1(1):1–20, 2011.

W. Lian, R. Henao, V. Rao, J. Lucas, and L. Carin. A multitask point process predictive model. In *International Conference on Machine Learning*, pp. 2030–2038. PMLR, 2015.

T. Long, F. Mehrdad, S. Le, and Z. Hongyuan. *NetCodec: Community Detection from Individual Activities*, pp. 91–99.

H. Mei, T. Wan, and J. Eisner. Noise-contrastive estimation for multivariate point processes. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 5204–5214. Curran Associates, Inc., 2020.

R. Näätänen and T. Picton. The n1 wave of the human electric and magnetic response to sound: A review and an analysis of the component structure. *Psychophysiology*, 24(4):375–425, 1987.

M. Okatan, M. A. Wilson, and E. N. Brown. Analyzing functional connectivity using a network likelihood model of ensemble neural spiking activity. *Neural computation*, 17(9):1927–1961, 2005.

J. W. Pillow, J. Shlens, L. Paninski, A. Sher, A. M. Litke, E. J. Chichilnisky, and E. P. Simoncelli. Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454 (7207):995–999, 2008.

Python Software Foundation. Python Language Reference, version 3.8. *http://python.org/*, 2019.

K. Rad and L. Paninski. Information rates and optimal decoding in large neural populations. *Advances in neural information processing systems*, 24:846–854, 2011.

M. A. Shafto, L. K. Tyler, M. Dixon, J. R. Taylor, J. B. Rowe, R. Cusack, A. J. Calder, W. D. Marslen-Wilson, J. Duncan, T. Dalgleish, et al. The cambridge centre for ageing and neuroscience (cam-can) study protocol: a cross-sectional, lifespan, multidisciplinary examination of healthy cognitive ageing. *BMC neurology*, 14(1):1–25, 2014.

O. Shchur, N. Gao, M. Biloš, and S. Günnemann. Fast and flexible temporal point processes with triangular maps. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 73–84. Curran Associates, Inc., 2020.

M. A. Sherman, S. Lee, R. Law, S. Haegens, C. A. Thorn, M. S. Hämäläinen, C. I. Moore, and S. R. Jones. Neural mechanisms of transient neocortical beta rhythms: Converging evidence from humans, computational modeling, monkeys, and mice. *Proceedings of the National Academy of Sciences*, 113(33):E4885–E4894, 2016.

C. Tallon-Baudry, O. Bertrand, C. Delpuech, and J. Pernier. Stimulus specificity of phase-locked and non-phase-locked 40 hz visual responses in human. *Journal of Neuroscience*, 16(13):4240–4249, 1996.

W. Truccolo, U. T. Eden, M. R. Fellows, J. P. Donoghue, and E. N. Brown. A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *Journal of neurophysiology*, 93(2):1074–1089, 2005.

P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C J Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17: 261–272, 2020.

I. Winkler, S. Debener, K.-R. Müller, and M. Tangermann. On the influence of high-pass filtering on ica-based artifact reduction in eeg-erp. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 4101–4105. IEEE, 2015.

B. Wohlberg. Convolutional sparse representation of color images. In *IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI)*, pp. 57–60, 2016a.

B. Wohlberg. Efficient algorithms for convolutional sparse representations. *Image Processing, IEEE Transactions on*, 25(1):301–315, 2016b.

H. Xu, M. Farajtabar, and H. Zha. Learning granger causality for hawkes processes. In *International conference on machine learning*, pp. 1717–1726, 2016.

A APPENDIX

A.1 DETAILS OF THE NEGATIVE LOG-LIKELIHOOD COMPUTATION

We defined the negative log-likelihood for parameters $\Theta_{k,\mathcal{P}} = (\mu_k, \boldsymbol{\alpha}_{k,\mathcal{P}}, \boldsymbol{m}_{k,\mathcal{P}}, \boldsymbol{\sigma}_{k,\mathcal{P}})$ as follow:

$$\mathcal{L}_{k,\mathcal{P}}(\Theta_{k,\mathcal{P}}) = \int_0^T \lambda_{k,\mathcal{P}}(t)\mathrm{d}t - \sum_{t\in\mathcal{A}_k} \log \lambda_{k,\mathcal{P}}(t) \ . \tag{16}$$

We show here in details that $\int_0^T \lambda_{k,p}(t)\mathrm{d}t = \mu_k T + \sum_{p\in\mathcal{P}} \alpha_{k,p} n_p$. Without loss of generality, we can first assume that $\forall p \in \mathcal{P}, t_{n_p}^{(p)} + b \leq T$. Hence,

$$\forall p \in \mathcal{P}, \forall i = 1, \ldots, n_p, \int_0^T \kappa_{k,p}\left(t - t_i^{(p)}\right) \mathrm{d}t = 1 \ .$$

Thus, we have that

$$\begin{aligned}
\int_0^T \lambda_{k,p}(t)\,\mathrm{d}t &= \int_0^T \mu_k + \sum_{p\in\mathcal{P}} \sum_{i, t_i^{(p)} < t} \alpha_{k,p}\kappa_{k,p}\left(t - t_i^{(p)}\right) \mathrm{d}t \\
&= \mu_k T + \sum_{p\in\mathcal{P}} \alpha_{k,p} \left(\sum_{t_i^{(p)} \in \mathcal{T}_p} \int_0^T \kappa_{k,p}\left(t - t_i^{(p)}\right) \mathrm{d}t \right) \\
&= \mu_k T + \sum_{p\in\mathcal{P}} \alpha_{k,p} n_p \ .
\end{aligned}$$

Finally, we have that the negative log-likelihood writes

$$\mathcal{L}_{k,\mathcal{P}}(\Theta_{k,\mathcal{P}}) = \mu_k T + \sum_{p\in\mathcal{P}} \alpha_{k,p} n_p - \sum_{t\in\mathcal{A}_k} \log \lambda_{k,\mathcal{P}}(t) \ .$$

A.2 DETAILS OF EM-BASED ALGORITHM COMPUTATIONS

First, recall that the negative log-likelihood for parameters $\Theta_{k,\mathcal{P}} = (\mu_k, \boldsymbol{\alpha}_{k,\mathcal{P}}, \boldsymbol{m}_{k,\mathcal{P}}, \boldsymbol{\sigma}_{k,\mathcal{P}})$ is as follow:

$$\mathcal{L}_{k,\mathcal{P}}\left(\Theta_{k,\mathcal{P}}\right) = \mu_k T + \sum_{p\in\mathcal{P}} \alpha_{k,p} n_p - \sum_{t\in\mathcal{A}_k} \log \left(\mu_k + \sum_{p\in\mathcal{P}} \sum_{i, t_i^{(p)} \leq t} \alpha_{k,p}\kappa_{k,p}\left(t - t_i^{(p)}; m_{k,p}, \sigma_{k,p}\right) \right) \ .$$

Recall also that we defined $P_k^{(n)}$ and $P_p^{(n)}$ as follow:

$$P_k^{(n)}(t) = \frac{\mu_k^{(n)}}{\lambda_{k,p}^{(n)}(t)} \qquad \text{and} \qquad \forall p \in \mathcal{P}, P_p^{(n)}(t) = \frac{\alpha_{k,p}^{(n)} \sum_{i, t_i^{(p)} \leq t} \kappa_{k,p}^{(n)}\left(t - t_i^{(p)}(t)\right)}{\lambda_{k,\mathcal{P}}^{(n)}(t)} \ , \tag{17}$$

where $\theta^{(n)}$ denotes the value of the parameter θ at step n of the algorithm, and similarly, if f is a function of parameter θ, $f^{(n)}(x;\theta) \coloneqq f(x;\theta^{(n)})$.

Bellow are details of calculation for parameters update for m and σ. We first rewrite the kernel function in order to have it under a form that simplifies further computations.

$$\begin{aligned}
\kappa(x; m, \sigma, a, b) &= \frac{1}{\sigma} \frac{\phi\left(\frac{x-m}{\sigma}\right)}{\Phi\left(\frac{b-m}{\sigma}\right) - \Phi\left(\frac{a-m}{\sigma}\right)} \mathbf{1}_{a\leq x\leq b} \\
&= \frac{\exp\left(-\frac{1}{2}\frac{(x-m)^2}{\sigma^2}\right)}{C\left(m, \sigma, a, b\right)} \mathbf{1}_{a\leq x\leq b}
\end{aligned}$$

where

$$C\left(m,\sigma,a,b\right) \coloneqq \int_a^b \exp\left(-\frac{1}{2}\frac{\left(u-m\right)^2}{\sigma^2}\right)\mathrm{d}u \ . \tag{18}$$

Hence, we can pre-compute its derivatives with respect to m and σ:

$$\frac{\partial}{\partial m}\kappa(x;m,\sigma,a,b) = \left(\frac{x-m}{\sigma^2} - \frac{C_m\left(m,\sigma,a,b\right)}{C\left(m,\sigma,a,b\right)}\right)\kappa(x;m,\sigma,a,b) \ , \tag{19}$$

and

$$\frac{\partial}{\partial \sigma}\kappa(x;m,\sigma,a,b) = \left(\frac{\left(x-m\right)^2}{\sigma^3} - \frac{C_\sigma\left(m,\sigma,a,b\right)}{C\left(m,\sigma,a,b\right)}\right)\kappa(x;m,\sigma,a,b) \ . \tag{20}$$

where $C_m\left(m,\sigma,a,b\right) \coloneqq \frac{\partial C}{\partial m}\left(m,\sigma,a,b\right)$ and $C_\sigma\left(m,\sigma,a,b\right) \coloneqq \frac{\partial C}{\partial \sigma}\left(m,\sigma,a,b\right)$.

Update equation for $m_{k,p}$

$$\begin{aligned}
\frac{\partial \mathcal{L}_{k,\mathcal{P}}}{\partial m_{k,p}}\left(\Theta_{k,\mathcal{P}}\right) &= -\sum_{t\in\mathcal{A}_k}\sum_{i,t_i^{(p)}\leq t}\left(\frac{t-t_i^{(p)}-m_{k,p}}{\sigma_{k,p}^2} - \frac{C_m\left(m_{k,p},\sigma_{k,p},a,b\right)}{C\left(m_{k,p},\sigma_{k,p},a,b\right)}\right)\frac{\alpha_{k,p}\kappa_{k,p}\left(t-t_i^{(p)}\right)}{\lambda_{k,\mathcal{P}}(t)} \\
&= \left(\frac{m_{k,p}}{\sigma_{k,p}^2} + \frac{C_m\left(m_{k,p},\sigma_{k,p},a,b\right)}{C\left(m_{k,p},\sigma_{k,p},a,b\right)}\right)\sum_{t\in\mathcal{A}_k}P_p(t) \\
&- \frac{1}{\sigma_{k,p}^2}\sum_{t\in\mathcal{A}_k}\sum_{i,t_i^{(p)}\leq t}\frac{\alpha_{k,p}\left(t-t_i^{(p)}\right)\kappa_{k,p}\left(t-t_i^{(p)}\right)}{\lambda_{k,\mathcal{P}}(t)}
\end{aligned} \tag{21}$$

Hence, by canceling the previous derivative,

$$m_{k,p}^{(n+1)} = \frac{\sum_{t\in\mathcal{A}_k}\sum_{i,t_i^{(p)}\leq t}\frac{\alpha_{k,p}^{(n)}\left(t-t_i^{(p)}\right)\kappa_{k,p}^{(n)}\left(t-t_i^{(p)}\right)}{\lambda_{k,\mathcal{P}}^{(n)}(t)}}{\sum_{t\in\mathcal{A}_k}P_p^{(n)}(t)} - {\sigma_{k,p}^{(n)}}^2\frac{C_m\left(m_{k,p}^{(n)},\sigma_{k,p}^{(n)},a,b\right)}{C\left(m_{k,p}^{(n)},\sigma_{k,p}^{(n)},a,b\right)} \tag{22}$$

Update equation for $\sigma_{k,p}$

$$\begin{aligned}
\frac{\partial \mathcal{L}_{k,\mathcal{P}}}{\partial \sigma_{k,p}}\left(\Theta_{k,\mathcal{P}}\right) &= -\sum_{t\in\mathcal{A}_k}\sum_{i,t_i^{(p)}\leq t}\left(\frac{\left(t-t_i^{(p)}-m_{k,p}\right)^2}{\sigma_{k,p}^3} - \frac{C_\sigma\left(m_{k,p},\sigma_{k,p},a,b\right)}{C\left(m_{k,p},\sigma_{k,p},a,b\right)}\right)\frac{\alpha_{k,p}\kappa_{k,p}\left(t-t_i^{(p)}\right)}{\lambda_{k,\mathcal{P}}(t)} \\
&= \frac{C_\sigma\left(m_{k,p},\sigma_{k,p},a,b\right)}{C\left(m_{k,p},\sigma_{k,p},a,b\right)}\sum_{t\in\mathcal{A}_k}P_p(t) \\
&- \frac{1}{\sigma_{k,p}^3}\sum_{t\in\mathcal{A}_k}\sum_{i,t_i^{(p)}\leq t}\frac{\alpha_{k,p}\left(t-t_i^{(p)}(t)-m_{k,p}\right)^2\kappa_{k,p}\left(t-t_i^{(p)}\right)}{\lambda_{k,\mathcal{P}}(t)}
\end{aligned} \tag{23}$$

Hence, by canceling the previous derivative,

$$\sigma_{k,p}^{(n+1)} = \left(\frac{C\left(m_{k,p}^{(n)},\sigma_{k,p}^{(n)},a,b\right)}{C_\sigma\left(m_{k,p}^{(n)},\sigma_{k,p}^{(n)},a,b\right)}\frac{\sum_{t\in\mathcal{A}_k}\sum_{i,t_i^{(p)}\leq t}\frac{\alpha_{k,p}^{(n)}\left(t-t_i^{(p)}(t)-m_{k,p}^{(n)}\right)^2\kappa_{k,p}^{(n)}\left(t-t_i^{(p)}\right)}{\lambda_{k,\mathcal{P}}^{(n)}(t)}}{\sum_{t\in\mathcal{A}_k}P_p^{(n)}(t)}\right)^{1/3} \tag{24}$$

Finally, to ensure that the σ coefficient stays strictly positive in order to avoid computational errors, we add a projection step onto $[\varepsilon\,;+\infty)$, with $\varepsilon > 0$: $\sigma_{k,p}^{(n+1)} = \pi_{[\varepsilon\,;+\infty)}\left(\sigma_{k,p}^{(n+1)}\right)$.

A.3 EXPERIMENTS ON SYNTHETIC DATA

In this section, we present figures that are complementary to the ones obtained on synthetic data, presented in subsection 4.1. In Figure A.1, we report the standard deviation associated with Figure 3, *i.e.,* the standard deviation of the relative infinite norm for different process duration T and different proportion of events kept P/S, for the two kernel shapes.

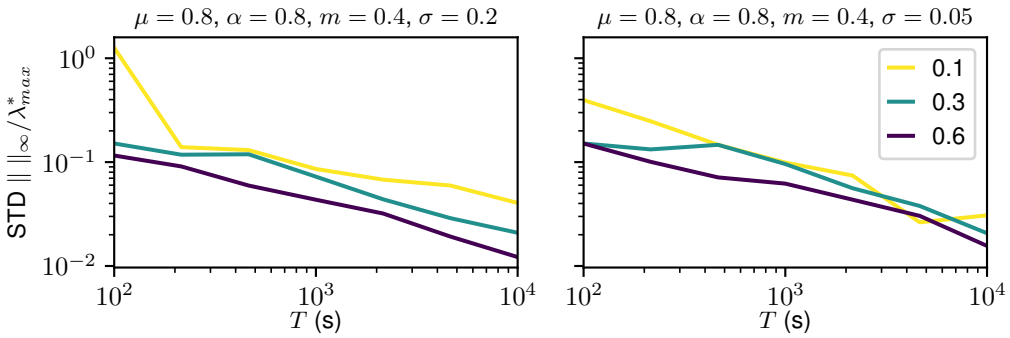

Figure A.1: Standard deviation of the relative infinite norm as a function of process duration T and the percentage of events kept P/S, for two kernel shapes on synthetic data: wide kernel (**left**) and sharp kernel (**right**). The variation of the EM estimates continually decreases with longer and denser processes.

Figure A.2 reports some details on the computational time of the experiment done on synthetic data which produced the Figure 3 and Figure A.1. For each value of T, the mean computational time is computed over 90 experiments (3 values of P/S times 30 random seeds). Results are presented in Figure A.2.

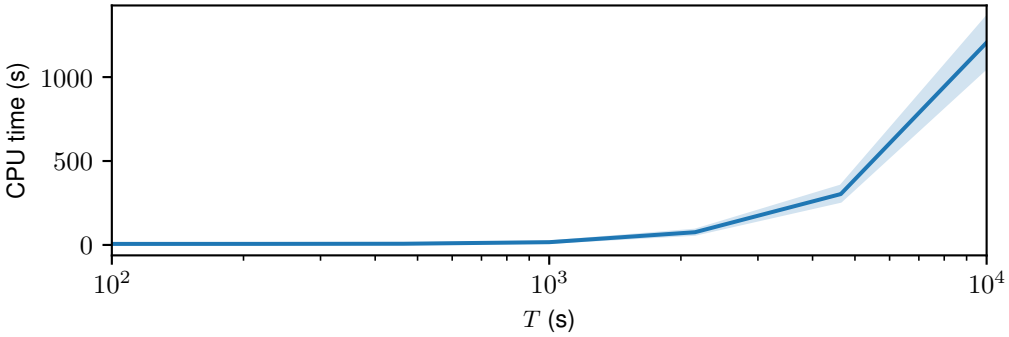

Figure A.2: Mean and 95 % CI of computational time (in seconds) for one EM algorithm, as a function of process duration T (in seconds), on synthetic data.

A.4 USUAL M/EEG DATA ANALYSIS

We present in this section some results obtained using usual M/EEG data analysis, such as Independent Component Analysis (ICA), epoch averaging, or time/frequency analysis. First, on MNE *sample* dataset, we proceed to an ICA in order to manually identify and remove artifacts. To do so, raw signals are filtered, and 15 independent components are fitted. The first two components, that we manually identify as corresponding to the eye-blink and heartbeat artifacts, are presented in Figure A.3. As mentioned, the ICA is commonly used to remove manually identified artifacts, in order to reconstruct the original signals free of those artifacts. However, there are two drawbacks of this method, the first one being that some domain related knowledge is needed in order to correcly identify the

artifacts. The second drawbacks happen when the signal is reconstructed after the removal of certain components. Indeed, such a reconstruction will lead to a loss of information across all channels, as the artifacts are shared by all the sensors. Thanks to the Convolutional dictionary learning (CDL) that extracts the different artifacts directly from the raw data, our method do not suffer of these drawbacks.

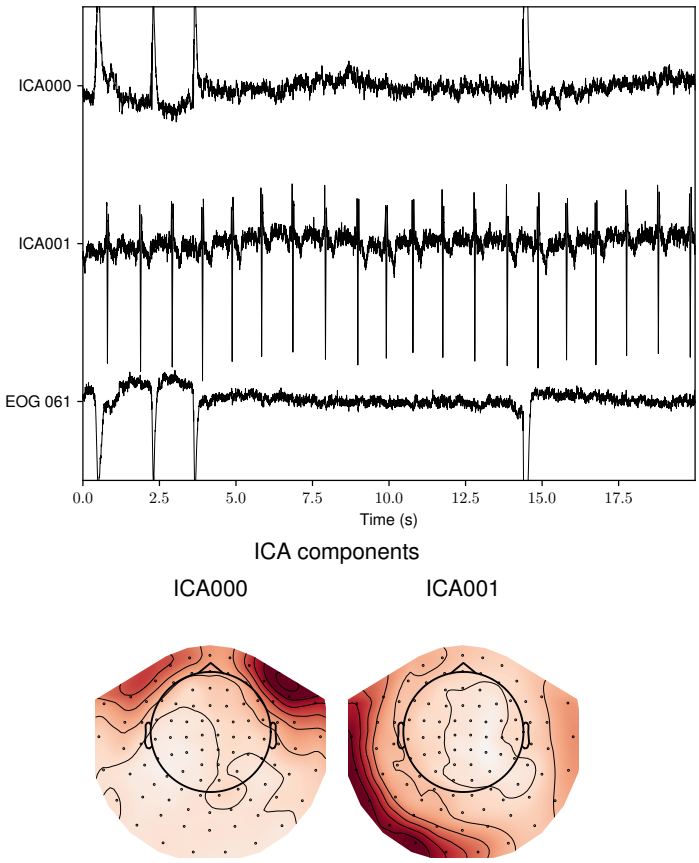

Figure A.3: The first two components alongside with electrooculography (EOG) signal (top) and their corresponding topomaps (bottom), on MNE *sample* dataset. `ICA000` can be associated to the eye-blink artifact, whereas `ICA001` can be associated to the heartbeat artifact.

Still on MNE *sample* dataset, we compute the epoch average following an auditory stimulus (stimuli in the left ear and in the right ear are combined) and plot on Figure A.4 the obtained evoked signals. Figure A.5 is similar but for the visual stimuli (again, stimuli on the left visual field and on the right visual field are combined).

On *somato* dataset, in order to exhibit the induced response to the somatosensory stimulus on the left median nerve of the subject, we perform a time/frequency analysis, presented on Figure A.6. In order to perform this analysis, a complex process including the use of Morlet wavelets.

Finally, note that these methods that are commonly used in the M/EEG data analysis comprise a thoughtfully data pre-processing as well as a manual intervention requiring domain knowledge in order to identify both artifacts and evoked and induced responses. As the proposed method in this paper is composed of a unified pipeline, it is a significant gain.

A.5 Experiments on MNE *somato* dataset

We present in this section extra results obtained on the real dataset *somato*, that are complementary of Figure 5. Figure A.7 shows 3 atoms that correspond all to a μ-wave located in the secondary

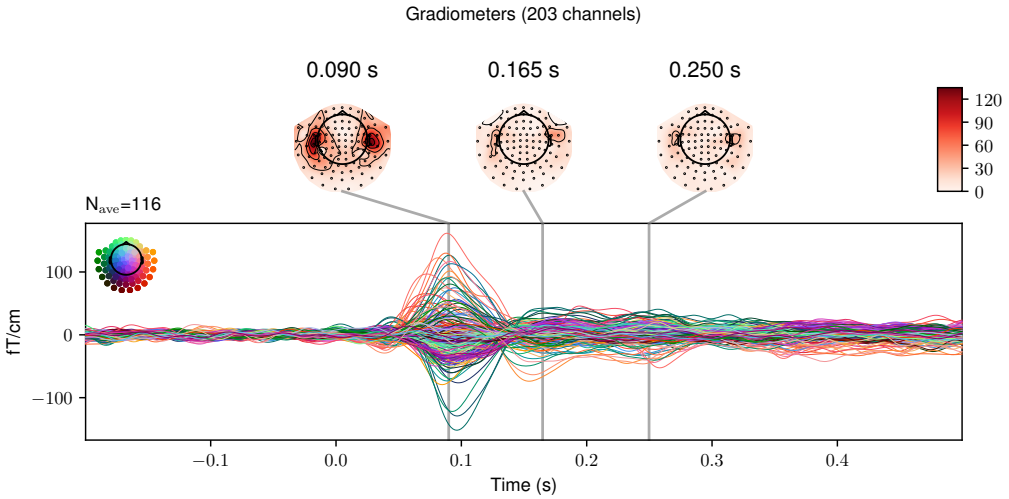

Figure A.4: Evoked signals following an auditory stimulus (cue at time = 0) on MNE *sample* dataset. Baseline correction applied from beginning of the data until time point zero.

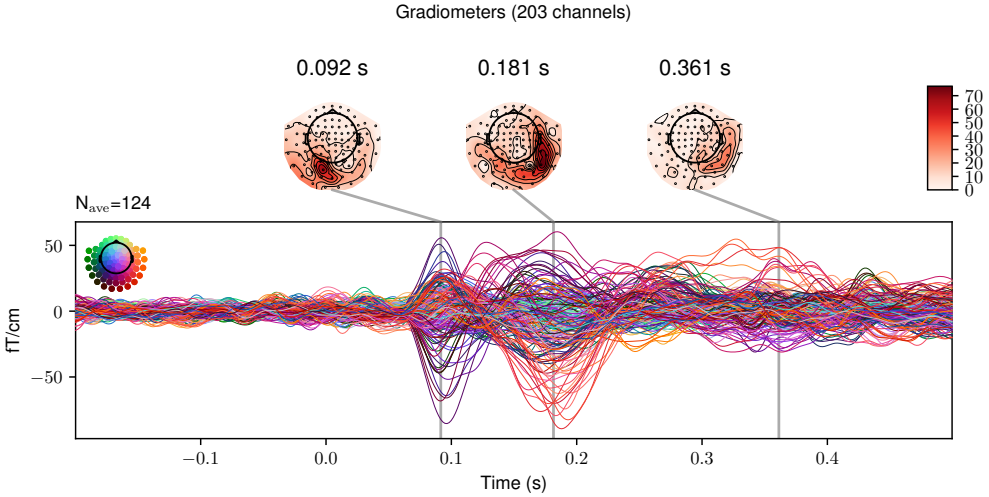

Figure A.5: Evoked signals following a visual stimulus (cue at time = 0) on MNE *sample* dataset. Baseline correction applied from beginning of the data until time point zero.

somatosensory region (S2), with three different shapes of kernel in their estimated intensity functions.

A.6 Experiments on Cam-CAN dataset

The Cam-CAN dataset contains data of M/EEG recording of 643 human subjects submitted to audio and visual stimuli. In this experiment, 120 bimodal audio/visual trials and eight unimodal trials – included to discourage strategic responding to one modality (four visual only and four auditory only) – are presented to each subject. For each bimodal trial, participants see two checkerboards presented to the left and right of a central fixation (34 milliseconds duration) and simultaneously hear a 300 milliseconds binaural tone at one of three frequencies (300, 600, or 1200 Hz, equal numbers of trials pseudorandomly ordered). For unimodal trials, participants either only hear a tone or see the

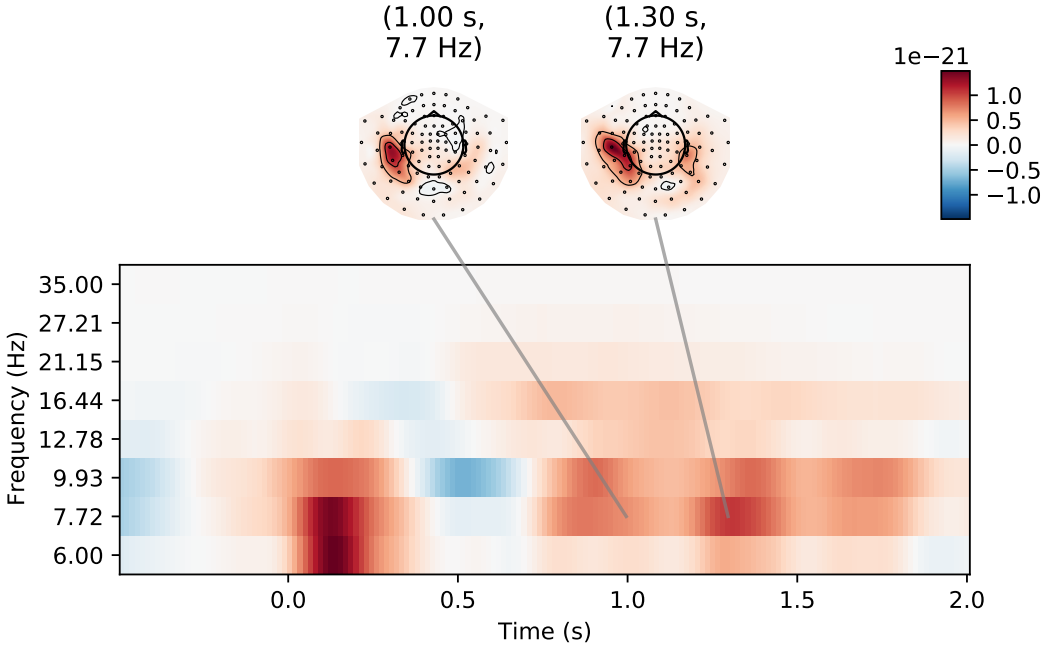

Figure A.6: Time-frequency plane for epoched signals following a somatosensory stimulus (cue at time = 0), on MNE *somato* dataset, with overviews at 1 and 1.3 seconds and 7.7 Hz. Baseline correction applied from time point -0.5 until time point zero.

checkerboards. For each trial, participants respond by pressing a button with their right index finger if they hear or see any stimuli (Shafto et al., 2014). For each subject, the experiment lasts less than $4\,\text{min}$.

The signals are pre-processed using low-pass filtering at $125\,\text{Hz}$ to remove slow drifts in the data, and are resampled to $150\,\text{Hz}$ to limit the atom size in the CDL. CDL is computed using `alphacsc` (Dupré la Tour et al., 2018) with the `GreedyCDL` method. 20 atoms of duration $0.5\,\text{s}$ are extracted from the signals. The extracted atoms'activations are binarized and similarly as previous experiments, the events time are shifted to make them correspond to the peak amplitude time in the atom. Then, for every atom, the intensity function is estimated using the EM-based algorithm with 200 iterations and the "smart start" initialization strategy. Kernels' truncation values are set at $[0\,\text{s}\,;0.9\,\text{s}]$. Two drivers per atom are considered. The first driver contains timestamps of all bimodal trials (all frequencies combined) with in addition the 4 auditory unimodal stimuli (denoted `catch0`). The second driver is similar to the first one, but instead of the auditory unimodal stimuli, it contains the 4 visual unimodal stimuli (denoted `catch1`).

This experiment is performed for 3 subjects, for each one we plot the 5 atoms that have the highest ratio α/μ, so that we automatically exhibit atoms that are highly linked to the presented stimuli. Similarly as results presented in Section 4, we plot the spatial and temporal representation of each atom, as well as the two learned intensities functions, that we plot "at kernel". Results are presented on Figure A.8, Figure A.9 and Figure A.10.

First, we can observe that, as for experiments on *sample* and *somato* datasets, we recover the heartbeat artifact (atom 0 in Figure A.9) as well as the eye-blink artifact (atoms 0 in Figure A.8 and in Figure A.10). Due to the experiment paradigm with in particular the instruction given to the subjects – that is to press a button after every stimuli, the majority of them being visual –, it is not surprising if the eye-blink artifact is slightly linked to the stimuli. Indeed, it is often observed in such experiments

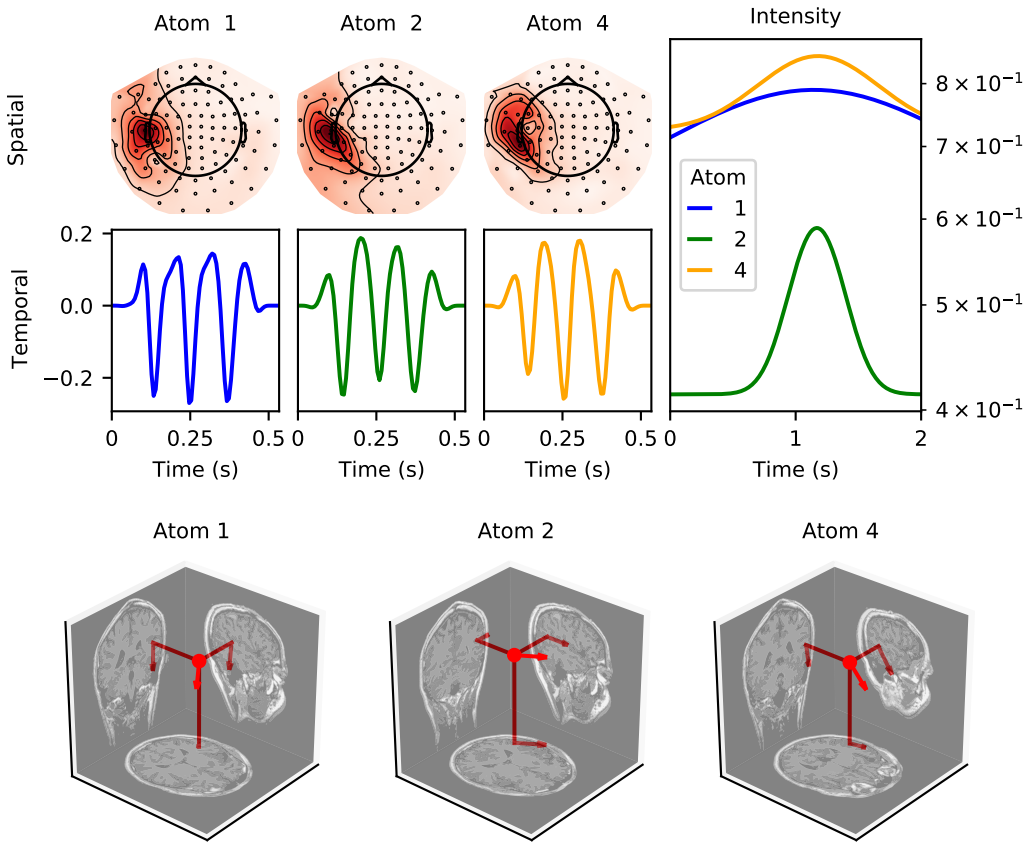

Figure A.7: Spatial and temporal patterns of 3 μ-wave atoms from *somato* dataset, alongside with their respective estimated intensity functions. Below, in order to provide further information, are the corresponding brain locations for each atom obtained with dipole fitting.

– where there is no designated time to blink – that the subject "allows" thyself to blink after the visual stimulus.

As the majority of the presented stimuli are a combination of visual and auditory, the CDL model struggles to separate the two corresponding neural responses. Hence, that is why it can be observed that most of the first atoms reported exhibit a mixture between auditory and visual response in their topographies (first row). However, one can observes that for some such atoms, DriPP learned intensity is able to indicate what is the main contributing stimulus in the apparition of the atom. For instance, for atom 10 in Figure A.8, the auditory stimulus is the main responsible stimulus of the presence of this atom, despite this latter presenting a mixture of both neural responses. A similar analysis can be made for atom 1 in Figure A.9, but this time for the auditory stimulus.

Finally, it is worth noting that these atoms highly depend on the CDL parameters. Indeed, with different parameters values – the number of atoms to extract, or for the duration of these atoms for instance –, it will lead to different results. This remark also applies to the previous experiments performed on MNE *sample* and *somato* datasets.

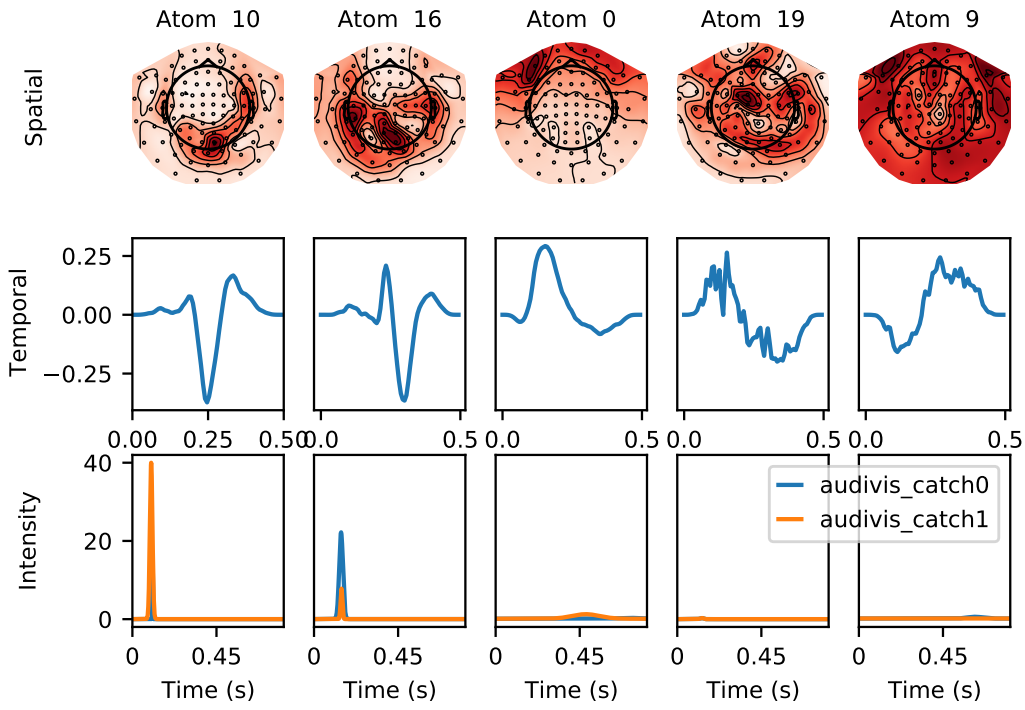

Figure A.8: Spatial and temporal patterns of the 5 atoms from Cam-CAN dataset subject CC620264, a 76.33 year old female, and their respective estimated intensity functions following a audiovisual stimulus (cue at time = 0 s). Atoms are ordered by their bigger ratio α/μ.

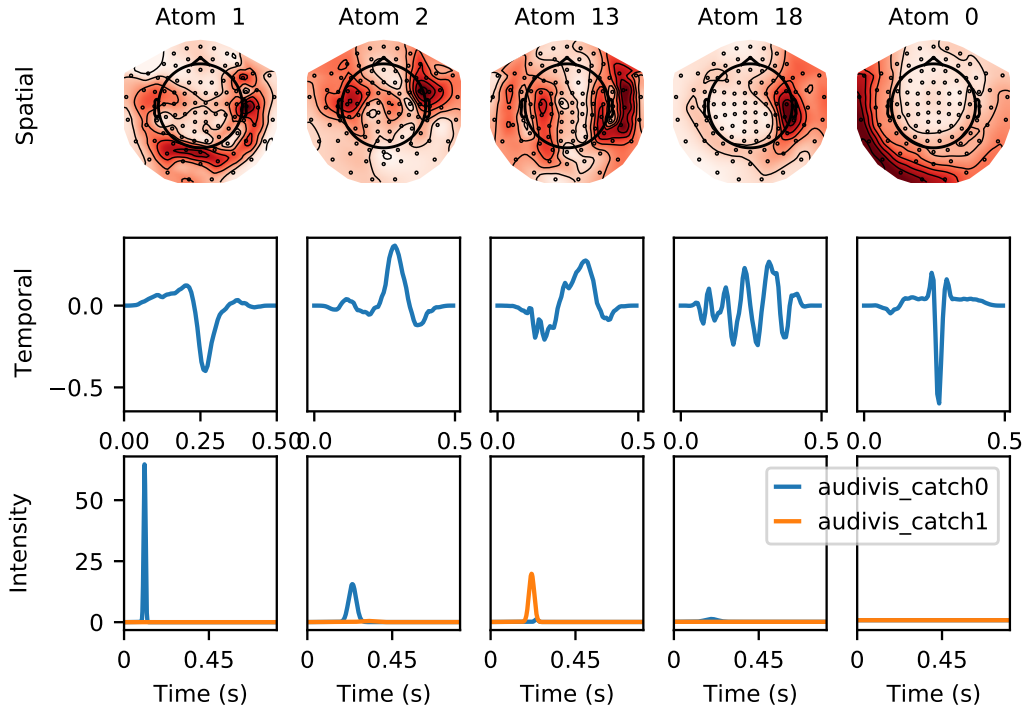

Figure A.9: Spatial and temporal patterns of the 5 atoms from Cam-CAN dataset subject CC520597, a 64.25 year old male, and their respective estimated intensity functions following a audiovisual stimulus (cue at time = 0 s). Atoms are ordered by their bigger ratio α/μ.

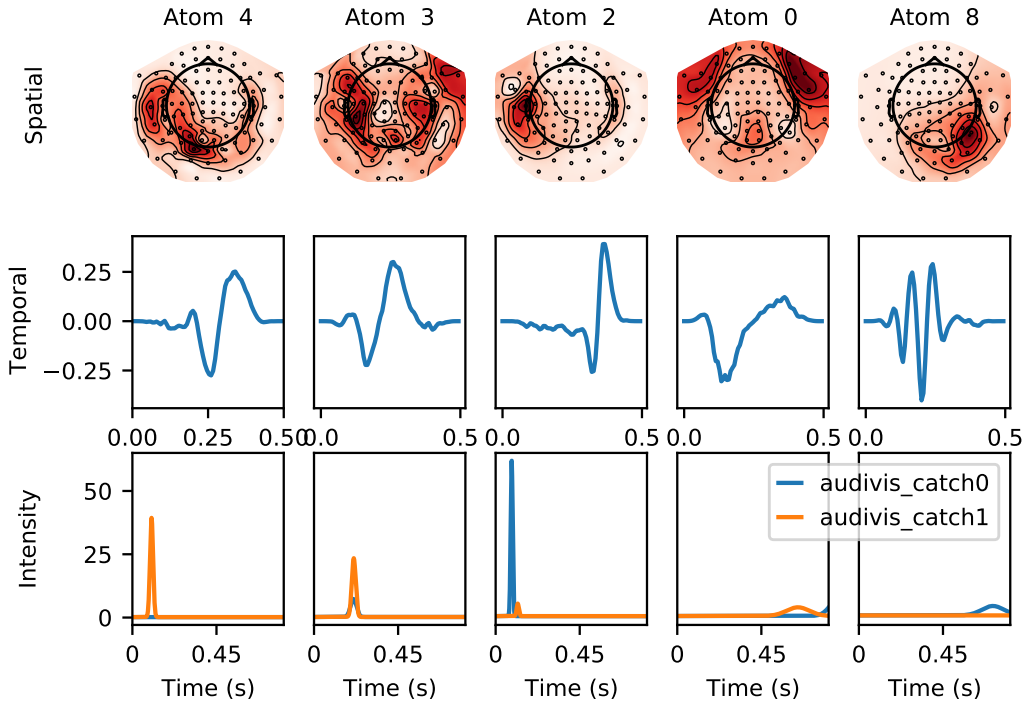

Figure A.10: Spatial and temporal patterns of the 5 atoms from Cam-CAN dataset subject CC723395, a86.08 year old female, and their respective estimated intensity functions following a audiovisual stimulus (cue at time = 0 s). Atoms are ordered by their bigger ratio α/μ.