entropy

MDPI

Article

Ant Colony System Optimization for Spatiotemporal Modelling of Combined EEG and MEG Data

Eugene A. Opoku [1,*], Syed Ejaz Ahmed [2], Yin Song [1] and Farouk S. Nathoo [1]

[1] Department of Mathematics and Statistics, University of Victoria, Victoria, BC V8P 5C2, Canada; yinsong@uvic.ca (Y.S.); nathoo@uvic.ca (F.S.N.)
[2] Department of Mathematics and Statistics, Brock University, St. Catharines, ON L2S 3A1, Canada; sahmed5@brocku.ca
* Correspondence: eopoku@uvic.ca

Abstract: Electroencephalography/Magnetoencephalography (EEG/MEG) source localization involves the estimation of neural activity inside the brain volume that underlies the EEG/MEG measures observed at the sensor array. In this paper, we consider a Bayesian finite spatial mixture model for source reconstruction and implement Ant Colony System (ACS) optimization coupled with Iterated Conditional Modes (ICM) for computing estimates of the neural source activity. Our approach is evaluated using simulation studies and a real data application in which we implement a nonparametric bootstrap for interval estimation. We demonstrate improved performance of the ACS-ICM algorithm as compared to existing methodology for the same spatiotemporal model.

Keywords: ant colony system; bayesian spatial mixture model; inverse problem; nonparamteric boostrap; EEG/MEG data

check for updates

Citation: Opoku, E.A.; Ahmed, S.E.; Song, Y.; Nathoo, F.S. Ant Colony System Optimization for Spatiotemporal Modelling of Combined EEG and MEG Data. *Entropy* **2021**, *23*, 329. https://doi.org/10.3390/e23030329

Academic Editor: David Holcman

Received: 21 January 2021
Accepted: 7 March 2021
Published: 11 March 2021

1. Introduction

Electroencephalography (EEG) and Magnetoencephalography (MEG) are two non-invasive approaches for measuring electrical activity of the brain with high temporal resolution. These neuroimaging techniques allow us to study brain dynamics and the complex informational exchange processes in the human brain. They are widely used in many clinical and research applications [1,2], though estimating the evoked-response activity within the brain from electromagnetic fields measured outside of the skull remains a challenging inverse problem with infinitely many different sources within the brain that can produce the same observed data [3].

Proposed solutions to the MEG/EEG inverse problem have been based on distributed source and dipolar methods [4]. In the case of distributed source methods, every location on a fine grid within the brain has associated neural activation source parameters. In this case, the number of unknown current sources exceeds the number of MEG or EEG sensors and estimation thus requires constraints through regularization or priors to obtain a solution. For such methods, various steps have been taken to regularize the solution by choosing minimum-norm solutions or by limiting the spatiotemporal variation of the solution. These approaches impose L_2 or L_1 [5] norm regularization constraints that serve to stabilize and condition the source parameter estimates. However, these methods do not consider the temporal nature of the problem. In [6], the authors propose a dynamic state-space model that accounts for both spatial and temporal correlations within and across candidate intra-cortical sources using Bayesian estimation and Kalman filtering. Dipolar methods, on the other hand, assume that the actual current distribution can be explained by a small set of current dipoles with unknown locations, amplitudes and orientations (see [4] for review). Hence, the resulting inverse problem becomes non-linear and a number of dipoles is to be estimated. Proposed solutions to this problem include algorithms such as simulated annealing [7] to address nonlinear optimization in the localization of neuromagnetic sources.

From the perspective of Bayesian approaches, the ill-posed nature of the inverse problem requires incorporation of prior assumptions when choosing an appropriate solution out of an infinite set of candidates. For instance, the authors of [8] consider Gaussian scale mixture models, with flexible, large covariance components representing spatial patterns of neural activity. The authors of [9] also propose a hierarchical linear model with Gaussian errors in a Parametric Empirical Bayes (PEB) framework whose random terms are drawn from multivariate Gaussian distributions and covariances factor into temporal and spatial components at the sensor and source levels. The authors of [10] propose an application of empirical Bayes to the source reconstruction problem with automatic selection of multiple cortical sources. The authors of [11] developed the Mesostate-Space Model (MSM) based on the assumption that the unknown neural brain activity can be specified in terms of a set of locally distributed and temporally coherent meso-sources for either MEG or EEG data, while the authors of [12] extend this approach to propose a Switching Mesostate-Space Model (SMSM) to allow flexibility by accounting for complex brain processes that cannot be characterized by linear and stationary Gaussian dynamics.

By extending and building on the MSM, the authors of [13] developed a Bayesian spatial finite mixture model incorporating the following two conditions, taken directly from [13]:

1. Relaxing the assumption of independent mixture allocation variables and modeling mixture allocations using the Potts model, which allows for spatial dependence in allocations.
2. Formulate the model for combined MEG and EEG data for joint source localization.

This spatiotemporal model describes a joint model that combines MEG and EEG data, in which brain neural activity is modeled from the Gaussian spatial mixture model. The neural source activity is described in terms of a few hidden states, with each state having its own dynamics and a Potts model used in representing the spatial dependence in the mixture model.

For the Bayesian mixture model formulated, an Iterated Conditional Modes (ICM) algorithm was developed by the authors of [13] for simultaneous point estimation and model selection for the number of mixture components in the latent process. Whilst ICM is a very simple and computationally efficient algorithm, convergence of this algorithm is sensitive to starting values and local optima. This issue was left unresolved in [13]. Here we investigate the potential for finding better solutions, and focus on implementing a population-based optimization algorithm-based Ant Colony System (ACS) [14] .

ACS is a metaheuristic optimization algorithm inspired by the biological behavior of ants constructing solutions based on their collective foraging behavior [14]. ACS has been successfully applied in several areas such as clustering, data mining and image segmentation problems [15–17]. ACS is a constructive algorithm that uses an analogue of ant trail pheromones to learn about good features of solutions in combinatorial optimization problems. New solutions are generated using a parameterized probabilistic model, the parameters of which are updated using previously generated solutions so as to direct the search towards promising areas of the solution space. The model used in ACS is known as pheromone, an artificial analogue of the chemical substance used by real ants to mark trails from the nest to food sources. Based on this representation, each artificial ant constructs a part of the solution based on concentration of pheromone information released by other ants. The amount of pheromone deposited by an ant reflects the quality of the good solutions built and the traversed path. The pheromone deposited and volatilized adds solution components to partial solutions. After some time and based on more ants' communications through pheromone information, they tend to follow the same optimal paths yielding the optimal solution, in our context maximization of the posterior density.

As an alternative to the ICM algorithm, we thus implement the ACS algorithm coupled with a local search ICM algorithm to provide a new approach to model estimation and potentially better estimates of the model parameters. This approach is evaluated and found to provide significant improvements. Within the context of a simpler spatial mixture, ACS has been implemented for a Gaussian Potts mixture model in [18] and has been shown to outperform both the Simulated Annealing and ICM algorithms for parameter and mixture component estimation. The theoretical guarantees associated with simulated annealing to reach a global optimum is dependent on the choice of a cooling schedule. The choice of an optimal cooling schedule can be difficult in practice for large spatiotemporal models. ACS has also proved to be competitive with genetic and other optimization algorithms in several tasks, mainly in image classification and the traveling salesman problem [19,20].

Ant Colony Optimization (ACO) algorithms are implemented to solve Constraint Satisfaction Problems (CSP) where ACO solutions to CSP face the challenge of high cost and low solution quality. Motivated by this challenge, the authors of [21] proposed Ant Colony Optimization based on information Entropy (ACOE). The idea is based on incorporating a local search that uses a crossover operation to optimize the best solution according to the feedback of information entropy. This is performed by comparing the difference of the information entropy between the current global best solution and the best solution in the current iteration. Datasets from four classes of binary CSP test cases were generated and then ACOE was implemented for comparison. Results showed that ACOE outperformed Particle Swarm Optimization (PSO), a Differential Evolution (DE) algorithm and Artificial Bee Colony (ABC) in terms of the solution quality, data distribution and convergence performance.

To our knowledge, this is the first attempt at solving the neuroelectromagnetic inverse problem for combined EEG/MEG data using a population-based optimization approach combined with a spatial mixture model. The primary contribution of this paper is the design and implementation of the ACS algorithm to the dynamic spatial model and its evaluation. Importantly, we demonstrate improved results both in the estimation of neural activity and model selection uniformly across all conditions considered.

The rest of the paper proceeds as follows. The posterior distribution of the model and the design and implementation of the ACS algorithm are presented in Section 2. In Section 3 our algorithm is investigated using simulation studies and comparisons made with an existing approach developed in [13] . Section 4 provides an illustration on real data and the development of a nonparametric bootstrap for interval estimation in a study of scrambled face perception. The paper concludes with a conclusion and directions for future work in Section 4.

Related Works

Merging EEG and MEG aims at accounting for information missed by one modality and captured by the other. Fused reconstruction therefore appears promising to reach high temporal and spatial resolutions in brain function imaging. The authors of [22] address the added value of combining EEG and MEG data for distributed source localization, building on the flexibility of parametric empirical Bayes, namely for EEG–MEG data fusion, group level inference and formal hypothesis testing. The proposed approach follows a two-step procedure by first using unimodal or multimodal inference to derive a cortical solution at the group level, and second by using this solution as a prior model for single subject-level inference based on either unimodal or multimodal data. Another popular approach for non-globally optimized solutions of the MEG/EEG inverse problem is based on the use of adaptive Beamformers (BF). However, the BFs are known to fail when dealing with correlated sources acting like poorly tuned spatial filters with a low signal-to-noise ratio (SNR) of the output time series and often meaningless cortical maps of power distribution. To address this limitation, the authors of [23] developed a novel data covariance approach to supply robustness to the beamforming technique when operating in an environment with correlated sources. To reduce the impact of the low spatial resolution of MEG and EEG, the authors of [24] developed a unifying framework for quantifying the spatial fidelity of MEG/EEG source estimates. This method quantifies the spatial fidelity of MEG/EEG estimates from simulated patch activations over the entire neocortex superposed on measured resting-state data. This approach grants more generalizability in the evaluation process that allows for, e.g., comparing linear and non-linear estimates in the whole brain for different Signal-to-Noise Ratios (SNR), number of active sources and activation waveforms. The authors of [25] discuss a solution to the source reconstruction problem and developed a novel hierarchical multiscale Bayesian algorithm for electromagnetic brain imaging using MEG and EEG within the context of sources that vary in spatial extent. In this Bayesian algorithm, the sensor data measurements are defined using a generative probabilistic graphical model that is hierarchical across spatial scales of brain regions and voxels. This algorithm enables robust reconstruction of sources that have different spatial extent, from spatially contiguous clusters of dipoles to isolated dipolar sources.

In [26], the authors propose a methodological framework for inverse-modeling of propagating cortical activity. Within this framework, cortical activity is represented in the spatial frequency domain, which is more natural than the dipole domain when dealing with spatially continuous activity. In dealing with multi-subject MEG/EEG source imaging, he authors of [27] propose a sparse multi-task regression that takes into account inter-subject variabilities known as the Minimum Wasserstein Estimates (MWE). This work jointly localizes sources for a population of subjects by casting the estimation as a multi-task regression problem in three key ideas. First, it proposes to use non-linear registration to obtain subject-specific lead field matrices that are spatially aligned. Second, it copes with the issue of inter-subject spatial variability of functional activations using optimal transport. Finally, it makes use of non-convex sparsity priors and joint inference of source estimates to obtain accurate source amplitudes. Various applications for MEG/EEG source reconstruction have been applied in the clinical setting for detection of epileptic spikes [28], identification of seizure onset zone [29] and presurgical workup of epilepsy patients [30–32].

2. Methods

This section describes the Bayesian spatial mixture model developed in [13] and the ACS-ICM algorithm.

2.1. Model

We provide details and mathematical description of the joint model below. Let $\boldsymbol{M}(t) = (M_1(t), M_2(t), ..., M_{n_M}(t))'$ and $\boldsymbol{E}(t) = (E_1(t), E_2(t), ..., E_{n_E}(t))'$ denote the MEG and EEG, respectively, at time t, $t = 1, \ldots, T$; where n_M and n_E denote the number of MEG and EEG sensors, the model assumes:

$$\begin{aligned} \boldsymbol{M}(t) &= \boldsymbol{X}_M \boldsymbol{S}(t) + \boldsymbol{\epsilon}_M(t), \quad \boldsymbol{\epsilon}_M(t) | \sigma_M^2 \overset{iid}{\sim} MVN(\boldsymbol{0}, \sigma_M^2 \boldsymbol{H}_M), \;\; t = 1, \ldots, T, \\ \boldsymbol{E}(t) &= \boldsymbol{X}_E \boldsymbol{S}(t) + \boldsymbol{\epsilon}_E(t), \quad \boldsymbol{\epsilon}_E(t) | \sigma_E^2 \overset{iid}{\sim} MVN(\boldsymbol{0}, \sigma_E^2 \boldsymbol{H}_E), \;\; t = 1, \ldots, T, \end{aligned}$$

where $\boldsymbol{X}_M$ and $\boldsymbol{X}_E$ denote $n_M \times P$ and $n_E \times P$ forward operators, respectively computed based on Maxwell's equations under the quasi-static assumption [33] for EEG and MEG; $\boldsymbol{H}_M$ and $\boldsymbol{H}_E$ are known $n_M \times n_M$ and $n_E \times n_E$ matrices, respectively, which can be obtained from baseline data providing information on the covariance structure of EEG and MEG sensor noise; and $\boldsymbol{S}(t) = (S_1(t), \ldots S_P(t))'$ represents the magnitude and polarity of neural currents sources over a fine grid covering the cortical surface. In this case, P represents a large number of point sources of potential neural activity within the brain covering the cortical surface. It is assumed that the P cortical locations are embedded in a 3D regular grid composed of N_v voxels to allow efficient computational implementation. Given this grid of voxels, a mapping $v : \{1, ..., P\} \to \{1, ..., N_v\}$ is defined such that $v(j)$ is the index of the voxel containing the jth cortical location. We assume a latent Gaussian mixture with allocations at the level of voxels:

$$S_j(t)|\boldsymbol{\mu}(t), \boldsymbol{\alpha}, \mathbf{Z} \overset{ind}{\sim} \prod_{l=1}^{K} N(\mu_l(t), \alpha_l)^{Z_{v(j)l}}, \tag{1}$$

$j = 1, \ldots, P,\ t = 1, \ldots, T$; where $\mathbf{Z} = (\mathbf{Z}_1', \mathbf{Z}_2', ..., \mathbf{Z}_{N_v}')'$ is a labeling process defined over the grid of voxels such that for each $v \in \{1, ..., N_v\}$, $\mathbf{Z}_v' = (Z_{v1}, Z_{v2}, ..., Z_{vK})$ with $Z_{vl} \in \{0, 1\}$ and $\sum_{l=1}^{K} Z_{vl} = 1$; $\boldsymbol{\mu}(t) = (\mu_1(t), \mu_2(t), ..., \mu_K(t))' = (\mu_1(t), \boldsymbol{\mu}^A(t)')'$, where $\boldsymbol{\mu}^A(t) = (\mu_2(t), ..., \mu_K(t))'$ denotes the mean of the "active" states over different components of activity and $\mu_1(t) = 0$ for all t, so that the first component corresponds to an "inactive" state. The variability of the l^{th} mixture component about its mean $\mu_l(t)$ is represented by $\alpha_l, l = 1, \ldots, K$.

The labeling process assigns each voxel to a latent state and is assumed to follow a Potts model:

$$P(\mathbf{Z}|\beta) = \frac{\exp\{\beta \sum_{h\sim j} \delta(\boldsymbol{Z_j}, \boldsymbol{Z_h})\}}{G(\beta)}, \quad \delta(\boldsymbol{Z_j}, \boldsymbol{Z_h}) = 2\boldsymbol{Z_j'}\boldsymbol{Z_h} - 1,$$

where $G(\beta)$ is the normalizing constant for this probability mass function, $\beta \geq 0$ is a hyper-parameter that governs the strength of spatial cohesion, and $i \sim j$ indicates that voxels i and j are neighbors, with a first-order neighborhood structure over the 3D regular grid. The mean temporal dynamics for active components is assumed to follow a first-order vector autoregressive process:

$$\boldsymbol{\mu}^A(t) = \boldsymbol{A}\boldsymbol{\mu}^A(t-1) + \mathbf{a}(t), \ \ \mathbf{a}(t)|\sigma_a^2 \overset{i.i.d}{\sim} MVN(\mathbf{0}, \sigma_a^2\mathbf{I})$$

$t = 2, \ldots, T$, $\boldsymbol{\mu}^A(1) \sim MVN(\mathbf{0}, \sigma_{\mu_1}^2\boldsymbol{I})$, with $\sigma_{\mu_1}^2$ fixed and known, but σ_a^2 unknown and assigned an inverse-Gamma (a_a, b_a) hyper-prior. Although in [13] a pseudo-likelihood approximation is adopted to the normalizing constant of the Potts model and then assigned a uniform prior to the spatial parameter to control the degree of spatial correlation, we fixed the inverse temperature parameter and vary it as part of a sensitivity analysis.

For model selection, the number of mixture components, the value of K, in Equation (1) will not be known prior and so it is estimated simultaneously with model parameters. Thus this approach achieves simultaneous point estimation and model selection. We can obtain a simple estimate for the number of mixture components based on the estimated allocation variables $\hat{\boldsymbol{Z}}$ when the algorithm is run with a sufficiently large value of K. This is achieved by running the algorithm with a value of K that is larger than the expected number of mixture components. For example, the value of K can be set as $K = 15$ when running the algorithm. The j^{th} location on the cortex is allocated to one of the mixture components based on the estimated value of $\hat{\mathbf{Z}}_{v(j)}$, where $\hat{\mathbf{Z}}_{v(j)} = (\hat{Z}_{v(j)_1}, \hat{Z}_{v(j)_2}, \ldots, \hat{Z}_{v(j)_K})'$ and $\hat{Z}_{v(j)_l} = 1$ if location j is allocated to component $l \in \{1, \ldots, K\}$. When the algorithm is run with a value of K that is large, there will result empty mixture components that have not been assigned any voxel locations under $\hat{\mathbf{Z}}$. In a sense these empty components have been automatically pruned out as redundant. The estimated number of mixture components can be obtained by counting the number of non-empty mixture components as follows:

$$\hat{K} = \sum_{l=1}^{K} I\{\sum_{v=1}^{n_v} \hat{Z}_{v_l} \neq 0\}.$$

This estimator requires us to run our algorithm only once for a single value of K and then the resulting number of mixture components assigned a location in $\hat{\boldsymbol{Z}}$ is determined and $\hat{K} \leq K$.

2.2. Ant Colony System

Ant Colony System (ACS) is a population-based optimization algorithm introduced in [14]. The basic structure of this algorithm is designed to solve the traveling salesman problem in which the aim is to find the shortest path to cover a given set of cities without revisiting any one of them. The inspiring source and development of this algorithm is the observation of the foraging behavior of real ants in their colony. This behavior is exploited in artificial ant colonies for the search of approximate solutions to discrete optimization problems, for continuous optimization problems, and for important problems in telecommunications, such as routing and load balancing, telecommunication network design, or problems in bioinformatics [34,35]. At the core of this algorithm is the communication between the ants by means of chemical pheromone trails, which enables them to collectively find short paths between their nest and food source. The framework of this algorithm can be categorized into four main parts: (1) construction of an agent ant solution, (2) local pheromone update of the solution, (3) improving solution by local search, and (4) global pheromone update of the best solution.

At each step of this constructive algorithm a decision is made concerning which solution component to add to the sequence of solution components already built. These decisions are dependent on the pheromone information, which represents the learned experience of adding a particular solution component given the current state of the solution under construction. The accumulated amount of pheromone mirrors the quality of the solution constructed based on the value of the objective function. The pheromone update aims to concentrate the search in regions of the search space containing high quality solutions while there is a stochastic component facilitating random exploration of the search space. In particular, the reinforcement of solution components depending on the solution quality is an important ingredient of ACS algorithms. To learn which components contribute to good solutions can help assembling them into better solutions. In general, the ACS approach attempts to solve an optimization problem by iterating the following two steps: (1) candidate solutions are constructed using a pheromone model, that is, a parameterized probability distribution over the solution space; (2) the candidate solutions are used to modify the pheromone values in a way that is deemed to bias future sampling toward high quality solutions.

The posterior distribution of the dynamic model takes the form $P(\boldsymbol{\Theta}|\boldsymbol{E},\boldsymbol{M}) = P(\boldsymbol{\Theta},\boldsymbol{E},\boldsymbol{M})/P(\boldsymbol{E},\boldsymbol{M})$, where:

$$\begin{aligned} P(\boldsymbol{\Theta},\boldsymbol{E},\boldsymbol{M}) &= P(\boldsymbol{E},\boldsymbol{M}|\boldsymbol{\Theta})P(\boldsymbol{\Theta}) = P(\boldsymbol{E}|\boldsymbol{\Theta})P(\boldsymbol{M}|\boldsymbol{\Theta})P(\boldsymbol{\Theta}) \\ &= \prod_{t=1}^{T} MVN(\boldsymbol{E}(t);\boldsymbol{X}_E\boldsymbol{S}(t),\sigma_E^2\boldsymbol{H}_E)\ \times MVN(\boldsymbol{M}(t);\boldsymbol{X}_M\boldsymbol{S}(t),\sigma_M^2\boldsymbol{H}_M) \\ &\times IG(\sigma_E^2;a_E,b_E)\times IG(\sigma_M^2;a_M,b_M)\times[\prod_{j=1}^{p}\prod_{t=1}^{T}\prod_{l=1}^{K}N(S_j(t);\mu_l(t),\alpha_l)^{Z_{v(j)l}}] \\ &\times[\prod_{t=2}^{T}MVN(\boldsymbol{\mu}^A(t);\boldsymbol{A}\mu^A(t-1),\sigma_a^2\boldsymbol{I})]\times MVN(\boldsymbol{\mu}^A(1);\boldsymbol{0},\sigma_{\mu_1}^2\boldsymbol{I})\times \text{Potts}(\boldsymbol{Z};\beta) \\ &\times\prod_{l=1}^{K}IG(\alpha_l;a_\alpha,b_\alpha)\times\times[\prod_{i=1}^{K-1}\prod_{j=1}^{K-1}N(A_{ij};0,\sigma_A^2)]\times IG(\sigma_a^2;a_a,b_a) \end{aligned} \tag{2}$$

where $MVN(\mathbf{x};\boldsymbol{\mu},\boldsymbol{\Sigma})$ denotes the density of the $\dim(\mathbf{x})$-dimensional multivariate normal distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ evaluated at $\mathbf{x}$; $IG(x;a,b)$ denotes the density of the inverse gamma distribution with parameters a and b evaluated at x; $N(x;\mu,\sigma^2)$ denotes the density of the normal distribution with mean μ and variance σ^2 evaluated at x; $\text{Potts}(\boldsymbol{Z};\beta)$ is the joint probability mass function of the Potts model with parameter β evaluated at $\boldsymbol{Z}$. Equation (2) represents the objective function to be maximized over $\boldsymbol{\Theta}$. The goal is to optimize over $\boldsymbol{\Theta} = \{\boldsymbol{S}(t),\boldsymbol{Z},\boldsymbol{\mu}(t),\boldsymbol{\alpha},\sigma_E^2,\sigma_M^2,\boldsymbol{A},\sigma_a^2\}$ maximizing the posterior (2).

ACS is based on set of agents, each representing artificial ants that construct solutions as sequences of solution components. Agent ant k builds a solution by allocating label ℓ from a set of voxel labels $\Lambda = \{1, \ldots K\}$ to the voxel $s \in \{1, ..., N_v\}$ based on a probabilistic transition rule $p^k(s, \ell)$. The transition rule quantifies the probability of ant k, assigning voxel s to label ℓ. This transition rule depends on the pheromone information $\tau(s, \ell)$ of the coupling (s, ℓ) representing the quality of assigning voxel s to label ℓ based on experience gathered by ants in the previous iteration. We let:

$$\ell = \begin{cases} \arg\max_u \tau(s, u) & \text{if } q \leq q_o \\ p^k(s, \ell) & \text{if } q > q_o \end{cases}$$

$$p^k(s, \ell) = \frac{\tau(s, \ell)}{\sum_{u \in \Lambda} \tau(s, u)} \tag{3}$$

where ℓ is a label for voxel s selected according to the transition rule above; $q \sim Uniform(0, 1)$; $q_o \in (0, 1)$ is a tuning parameter. An artificial ant chooses, with probability q_0, the solution component that maximizes the pheromone function $\tau(s, \ell)$ or it performs, with probability $1 - q_0$, a probabilistic construction step according to (3). The ACS pheromone system consists of two update rules; one rule is applied whilst constructing solutions (local pheromone update rule) and the other rule is applied after all ants have finished constructing a solution (global pheromone update rule). After assigning a label to a voxel, an ant modifies the amount of pheromone of the chosen couples (s, ℓ) by applying a local pheromone update (4):

$$\tau(s, \ell) \leftarrow (1 - \rho)\tau(s, \ell) + \rho\tau_o \tag{4}$$

where $\rho \in (0, 1)$ is a tuning parameter that controls evaporation of the pheromone and τ_o is the initial pheromone value. This operation simulates the natural process of pheromone evaporation preventing the algorithm from converging too quickly (all ants constructing the same solution) and getting trapped into a poor solution. In practice, the effect of this local pheromone update is to decrease the pheromone values via evaporation $(1 - \rho)\tau(s, \ell)$ on the visited solution components, making these components less desirable for the subsequents ants. The value of the evaporation rate indicates the relative importance of the pheromone values from one iteration to the following one. If ρ takes a value near 1, then the pheromone trail will not have a lasting effect, and this mechanism increases the random exploration of the search space within each iteration and helps avoid a too rapid convergence of the algorithm toward a sub-optimal region of the parameter space, whereas a small value will increase the importance of the pheromone, favoring the exploitation of the search space near the current solution.

To improve all solutions constructed and also update the other model parameters, we considered incorporating ICM as a local search method. Here, the ICM algorithm is used for both updating the model parameters and also for a local search over the mixture allocation variables. Thus, the update steps corresponding to ACS are combined with running ICM to convergence at each iteration. Finally, after all solutions have been constructed by combined ACS and ICM steps, the quality of all solutions is evaluated using the objective function where the corresponding best solution is selected. We use a global update rule, where pheromone evaporation is again applied on the best solution chosen. Assuming voxel j is assigned to label v for the best solution, the global update is given as:

$$\tau(j, v) \leftarrow \begin{cases} (1 - \rho)\tau(j, v) + \rho\tau_o, & \\ (1 - \rho)\tau(j, k), & \text{and for all } k \neq v \end{cases}$$

The steps described are performed repeatedly until a change in the objective function becomes negligible and the model parameters from the best solution are returned as the final parameter estimates. The optimal values for the tuning parameters (q_o, τ_o, ρ) used in our ACS-ICM algorithm depend on the data. The strategy we adopt for choosing the tuning parameters is by using an outer level optimization on top of the ACS-ICM algorithm to optimize over tuning parameters (q_o, τ_o, ρ) within updates at the outer level based on the Nelder–Mead algorithm [36] applied to optimize over tuning parameters.

In order to reduce the dimension of parameter space and computing time, we apply clustering to the estimated neural sources. This is achieved by implementing a K-means algorithm to cluster the P locations on the cortex into a smaller number of $J \leq P$ clusters, assuming that $S_j(t) = S_l(t)$ for cortical locations l, j belonging to the same cluster. We investigated different values of $J = 250, 500, 1000$ in our simulation studies. Within the ICM algorithm, the labeling process $\boldsymbol{Z}$ is updated using an efficient chequerboard updating scheme [13]. The update scheme starts with partitioning $\boldsymbol{Z}$ into two blocks $\boldsymbol{Z} = \{\boldsymbol{Z}_W, \boldsymbol{Z}_B\}$ based on a three-dimensional chequerboard arrangement, where $\boldsymbol{Z}_W$ corresponds to "white" voxels and $\boldsymbol{Z}_B$ corresponds to "black" voxels. Under the Markov random field prior with a first-order neighborhood structure, the elements of $\boldsymbol{Z}_W$ are conditionally independent given $\boldsymbol{Z}_B$, the remaining parameters, and the data ***E*, *M***. This allows us to update $\boldsymbol{Z}_W$ in a single step, which involves simultaneously updating its elements from their full conditional distributions. The variables $\boldsymbol{Z}_B$ are updated in the same way.

It is well-known that the ICM algorithm is sensitive to initial values and the authors of [13] found this to be the case with the ICM algorithm developed for the spatiotemporal mixture model. The solution obtained, and even the convergence of the algorithm depend rather heavily on the starting values chosen. In the case of ACS-ICM, regardless of the initial values, the algorithm finds a better solution with the optimal tuning parameters and this solution tends to be quite stable. This is because ACS-ICM is a stochastic search procedure in which the pheromone update concentrates the search in regions of the search space containing high quality solutions to reach an optimum. When considering a stochastic optimization algorithm, there are at least two possible types of convergence that can be considered: convergence in value and convergence in solution. With convergence in value, we are interested in evaluating the probability that the algorithm will generate an optimal solution at least once. On the contrary, with convergence in solution we are interested in evaluating the probability that the algorithm reaches a state that keeps generating the same optimal solution. The convergence proofs are presented in [37,38]. The authors of [37] proved convergence with a probability of $1 - \epsilon$ for the optimal solution and more in general for any optimal solution in [38] of the ACS algorithm. This supports the argument that theoretically the application of ACS-ICM to source reconstruction should improve ICM .

The local search ICM algorithm procedure is presented in Algorithm 1 and the ACS-ICM algorithm is presented in Algorithm 2. Convergence of the ICM algorithms is monitored by examining the relative change of the Frobenius norm of the estimated neural sources on consecutive iterations.

Algorithm 1 presents a detailed description of the ICM algorithm. The ICM algorithm requires full conditional distributions of each model parameter where the mode of the distribution is taken as the update step for the parameter. The full conditional distribution are described and presented in [13]. This ICM algorithm is embedded in our ACS-ICM algorithm.

Algorithm 1 Iterated Conditional Modes (ICM) Algorithm.

```
$\Theta = \{\boldsymbol{S}(t), \boldsymbol{Z}, \boldsymbol{\mu}(t), \boldsymbol{\alpha}, \sigma_E^2, \sigma_M^2, \boldsymbol{A}, \sigma_a^2\} \leftarrow$ Initial Value
Converged $\leftarrow 0$
while Converged $= 0$ do
    $\sigma_M^2 \leftarrow \Big[\sum_{t=1}^{T} \frac{1}{2}\big(\boldsymbol{M}(t) - \boldsymbol{X}_M\boldsymbol{S}(t)\big)'\boldsymbol{H}_M^{-1}\big(\boldsymbol{M}(t) - \boldsymbol{X}_M\boldsymbol{S}(t)\big) + b_M\Big] / \Big[a_M + \frac{TN_M}{2} + 1\Big]$
    $\sigma_E^2 \leftarrow \Big[\sum_{t=1}^{T} \frac{1}{2}\big(\boldsymbol{E}(t) - \boldsymbol{X}_E\boldsymbol{S}(t)\big)'\boldsymbol{H}_E^{-1}\big(\boldsymbol{E}(t) - \boldsymbol{X}_E\boldsymbol{S}(t)\big) + b_E\Big] / \Big[a_E + \frac{TN_E}{2} + 1\Big]$
    $\sigma_a^2 \leftarrow \Big[\sum_{t=2}^{T} \frac{1}{2}(\boldsymbol{\mu}^A(t) - \boldsymbol{A}\boldsymbol{\mu}^A(t-1))'(\boldsymbol{\mu}^A(t) - \boldsymbol{A}\boldsymbol{\mu}^A(t-1)) + b_a\Big] / \Big[a_a + \frac{(T-1)(K-1)}{2} + 1\Big]$
    $vec(\boldsymbol{A}) \leftarrow \left(\frac{1}{\sigma_a^2}\left(\sum_{t=2}^{T}\boldsymbol{\mu}^A(t)'\boldsymbol{Kr_t}\right) \times \boldsymbol{C}_1^{-1}\right)'$, where $\boldsymbol{C}_1 = \frac{1}{\sigma_A^2}\boldsymbol{I}_{(K-1)^2} + \frac{1}{\sigma_a^2}\left(\sum_{t=2}^{T}\boldsymbol{Kr_t}'\boldsymbol{Kr_t}\right)$,
    and $\boldsymbol{Kr_t} = \left(\boldsymbol{\mu}^A(t-1)' \otimes \boldsymbol{I}_{K-1}\right)$
    for $l = 1, ..., K$ do
        $\alpha_l \leftarrow \Big[\frac{\sum\limits_{j=1}^{P}\sum\limits_{t=1}^{T} Z_{v(j)l}(S_j(t) - \mu_l(t))^2}{2} + b_\alpha\Big] / \Big[\frac{T\sum\limits_{j=1}^{P} Z_{v(j)l}}{2} + a_\alpha + 1\Big]$
    end for
    $\boldsymbol{\mu}(1) \leftarrow \left(\left(\sum_{j=1}^{P}(S_j(1)\vec{I}_{K-1})'\boldsymbol{D}_j + \frac{1}{\sigma_a^2}\boldsymbol{\mu}^A(2)'\boldsymbol{A}\right) \times \boldsymbol{B}_1^{-1}\right)'$, where $\boldsymbol{B}_1 = \sum_{j=1}^{P}\boldsymbol{D}_j + \frac{1}{\sigma_a^2}\boldsymbol{A'A} + \frac{1}{\sigma_{\mu_1}^2}\boldsymbol{I}_{K-1}$, $\boldsymbol{D}_j = \mathrm{Diag}(\frac{Z_{v(j)l}}{\alpha_l}, l =$
    $2, ..., K)$, $\vec{I}_{K-1} = (1, 1, \ldots, 1)'$ with dim $(\vec{I}_{K-1}) = K - 1$
    for $t = 2, ..., T-1$ do
        $\boldsymbol{\mu}(t) \leftarrow \left(\left(\sum_{j=1}^{P}(S_j(t)\vec{I}_{K-1})'\boldsymbol{D}_j + \frac{1}{\sigma_a^2}(\boldsymbol{\mu}^A(t+1))'\boldsymbol{A} + \frac{1}{\sigma_a^2}(\boldsymbol{\mu}^A(t-1)'\boldsymbol{A}')\right) \times \boldsymbol{B}_2^{-1}\right)'$
        where $\boldsymbol{B}_2 = \sum_{j=1}^{P}\boldsymbol{D}_j + \frac{1}{\sigma_a^2}(\boldsymbol{A'A} + \boldsymbol{I}_{K-1})$
    end for
    $\boldsymbol{\mu}(T) \leftarrow \left(\left(\sum_{j=1}^{P}(S_j(T)\vec{I}_{K-1})'\boldsymbol{D}_j + \frac{1}{\sigma_a^2}(\boldsymbol{\mu}^A(T-1)'\boldsymbol{A}')\right) \times \boldsymbol{B}_3^{-1}\right)'$
    where $\boldsymbol{B}_3 = \sum_{j=1}^{P}\boldsymbol{D}_j + \frac{1}{\sigma_a^2}\boldsymbol{I}_{k-1}$
    for $j = 1, ..., P$ do
        $\boldsymbol{S}_j \leftarrow -\frac{1}{2}\boldsymbol{\Sigma}_{S_j}\boldsymbol{W}_{2j}$        ▷ $\boldsymbol{S}_j = (S_j(1), S_j(2), ..., S_j(T))'$
        $\boldsymbol{\Sigma}_{S_j}^{-1} = W_{1j}\boldsymbol{I}_T, \boldsymbol{W}_{2j}' = (W_{2j}(1), W_{2j}(2), ..., W_{2j}(T))$
        where $W_{1j} = \frac{1}{\sigma_M^2}\left(\boldsymbol{X}_M[,j]'\boldsymbol{H}_M^{-1}\boldsymbol{X}_M[,j]\right) + \frac{1}{\sigma_E^2}\left(\boldsymbol{X}_E[,j]'\boldsymbol{H}_E^{-1}\boldsymbol{X}_E[,j]\right) + \sum_{l=1}^{K}\frac{Z_{v(j)l}}{\alpha_l}$
        $W_{2j}(t) = \frac{1}{\sigma_M^2}\Big(-2\boldsymbol{M}(t)'\boldsymbol{H}_M^{-1}\boldsymbol{X}_M[,j] + 2(\sum_{v \neq j}\boldsymbol{X}_M[,v]S_v(t))'\boldsymbol{H}_M^{-1}\boldsymbol{X}_M[,j]\Big)$
        $+ \frac{1}{\sigma_E^2}\Big(-2\boldsymbol{E}(t)'\boldsymbol{H}_E^{-1}\boldsymbol{X}_E[,j] + 2(\sum_{v \neq j}\boldsymbol{X}_E[,v]S_v(t))'\boldsymbol{H}_E^{-1}\boldsymbol{X}_E[,j]\Big) - 2\sum_{l=1}^{K}\frac{\mu_l(t)}{\alpha_l}$
        $\boldsymbol{X}_M[,j]$, $\boldsymbol{X}_E[,j]$ denote the $jth$ column of $\boldsymbol{X}_E$ and $\boldsymbol{X}_M$
    end for
    Let $\mathbb{B}$ denote the indices for "black" voxels and $\mathbb{W}$ denote the indices for "white" voxels.
```

Algorithm 1 *Cont.*

```
    for $\kappa \in \mathbb{B}$ simultaneously do
        $Z_{\kappa q} \leftarrow 1$ and $Z_{\kappa l} \leftarrow 0,\ \forall l \neq q$
    where $q = \text{argmax}_{h \in \{1,\ldots,K\}} P(h)$, and
        $P(h) = \frac{\alpha_h^{-TN_{j\kappa}/2} \times \exp\left(-\frac{1}{2}\sum_{j|v(j)=\kappa} \alpha_h^{-1} \sum_{t=1}^{T}(S_j(t)-\mu_h(t))^2 + 2\beta\sum_{v\in\delta_\kappa} Z_{vh}\right)}{\sum_{l=1}^{K} \alpha_l^{-TN_{j\kappa}/2} \times \exp\left(-\frac{1}{2}\sum_{j|v(j)=\kappa} \alpha_l^{-1} \sum_{t=1}^{T}(S_j(t)-\mu_l(t))^2 + 2\beta\sum_{v\in\delta_\kappa} Z_{vl}\right)}$
    where $N_{j\kappa}$ is the number of cortical locations contained in voxel $\kappa$.
    end for
    for $\kappa \in \mathbb{W}$ simultaneously do
        $Z_{\kappa q} \leftarrow 1$ and $Z_{\kappa l} \leftarrow 0,\ \forall l \neq q$
    where $q = \text{argmax}_{h \in \{1,\ldots,K\}} P(h)$, and
        $P(h) = \frac{\alpha_h^{-TN_{j\kappa}/2} \times \exp\left(-\frac{1}{2}\sum_{j|v(j)=\kappa} \alpha_h^{-1} \sum_{t=1}^{T}(S_j(t)-\mu_h(t))^2 + 2\beta\sum_{v\in\delta_\kappa} Z_{vh}\right)}{\sum_{l=1}^{K} \alpha_l^{-TN_{j\kappa}/2} \times \exp\left(-\frac{1}{2}\sum_{j|v(j)=\kappa} \alpha_l^{-1} \sum_{t=1}^{T}(S_j(t)-\mu_l(t))^2 + 2\beta\sum_{v\in\delta_\kappa} Z_{vl}\right)}$
    where $N_{j\kappa}$ is the number of cortical locations contained in voxel $\kappa$.
    end for
    Check for convergence. Set Converged = 1 if so.
end while
```

Algorithm 2 Ant Colony System (ACS)-ICM Algorithm.

1. $\Theta \longleftarrow$ Initial Value; set tuning parameters τ_o, q_o, ρ and N_{ants}.
2. Initialize pheromone information $\tau(i, \ell) = \tau_o$, for each $(i, \ell) \in \{1, \ldots N_v\} \times \{1, \ldots K\}$ representing information gathered by ants.
3. Construct candidate solutions for each of N_{ants} ants. For ant j, we find a candidate voxel labeling $\mathbf{Z}^{(j)} = (\mathbf{Z}_1^{\prime(j)}, \mathbf{Z}_2^{\prime(j)}, \ldots, \mathbf{Z}_{N_v}^{\prime(j)})'$. This is done sequentially for each ant j.

    - Construct candidate by assigning label l to voxel s using the transition probability rule:

    $$\ell = \begin{cases} \arg\max_u \tau(s,u) & \text{if } q \leq q_o \\ p(s,\ell) & \text{if } q > q_o \end{cases}$$

    where if $q > q_o$ the label for voxel s is drawn randomly from $\{1, \ldots, K\}$ with probability

    $$p(s,\ell) = \frac{\tau(s,\ell)}{\sum_{u\in\Lambda} \tau(s,u)},$$

    and where $q \sim \text{uniform}[0,1]$.

    - Assuming voxel s is assigned label ℓ set:

    $$\tau(s,\ell) \leftarrow (1-\rho)\tau(s,\ell) + \rho\tau_o$$

    and for all $k \neq l$

    $$\tau(s,\ell) \leftarrow (1-\rho)\tau(s,k)$$

    where ρ is a tuning parameter in (0,1), which represents evaporation of the pheromone trails and $\tau_o > 0$.

Algorithm 2 *Cont.*

```
4: For all ants, improve candidate solutions by running ICM to convergence (this also allows an update to the other model parameters)
   $\boldsymbol{\Theta} = \{\{\boldsymbol{\mu}^A(1), \boldsymbol{\mu}^A(2), ..., \boldsymbol{\mu}^A(T)\}, \{\alpha_1, \alpha_2, ..., \alpha_k\}, \sigma_E^2, \sigma_M^2, \{S_j(t), t = 1, 2 \ldots, T, j = 1, 2, ..., P\}, \boldsymbol{A}, \sigma_a^2\}$.
5: For all $N_{ants}$ solutions, evaluate the quality of each ant's solution using objective function: $P(\boldsymbol{\Theta}, \boldsymbol{E}, \boldsymbol{M})$. Keep track of the best value.
   The current solution for each ant serves as the starting value for the next iteration.
6: Apply global updating of the pheromone function. For the best solution, $(s, \ell)$ update the pheromone as follows:
       Assuming voxel $s$ is assigned label $\ell$ set:
           $\tau(s, \ell) \leftarrow (1 - \rho)\tau(s, \ell) + \rho\tau_o$
       and for all $k \neq \ell$:
           $\tau(s, \ell) \leftarrow (1 - \rho)\tau(s, k)$
       Check for convergence via increase in $logP(\boldsymbol{\Theta}, \boldsymbol{E}, \boldsymbol{M})$. Go back to step 3
7: Return all voxel labeling $\mathbf{Z}$ and model parameters $\boldsymbol{\Theta}$ from the best solution.
```

3. Simulation Studies

In this section, we use a simulation study to evaluate the performance of our algorithm. The simulation study assesses the quality of the source estimates and the optimized objective function values obtained when using our proposed algorithm in comparison to the existing ICM algorithm developed in [13]. We then make comparisons between ACS-ICM and the ICM algorithm applied to combined simulated EEG and MEG data.

3.1. Proposed Approach

The MEG and EEG data were both generated from four scenarios with two, three, four and nine latent states corresponding to regions of neural activity. In each of the four cases, one of the states is inactive, while the remaining states represent different regions of brain activity generated by Gaussian signals. The temporal profile of brain activity at each of the brain locations in the activated regions is depicted in Appendix A, Figures A1 and A2. We projected the source activity at 8196 brain locations from the cortex onto the MEG and EEG sensor arrays using the forward operators $\boldsymbol{X}_M$ and $\boldsymbol{X}_E$. The simulated data were then obtained by adding Gaussian noise at each sensor, where the variance of the noise at each sensor was set to be 5% of the temporal variance of the signal at that sensor. The number of mixture components K was set to be the true number of latent states (either two, three, four, or nine) in the model. We simulated 500 replicate datasets and both ACS-ICM and ICM were applied to each dataset. For each simulated dataset we applied our algorithm with J = 250, 500, 1000 clusters so as to evaluate how the performance varies as this tuning parameter changes. We initialized both algorithms using the same starting values. For each replicate we computed the correlation between the estimated sources and the true sources $\text{Corr}[(\boldsymbol{S}(1)', \boldsymbol{S}(2)', \ldots, \boldsymbol{S}(T)'), (\hat{\boldsymbol{S}}(1)', \hat{\boldsymbol{S}}(2)', \ldots, \hat{\boldsymbol{S}}(T)')]$ as a measure of agreement. This measure was also averaged over the 500 replicate datasets to compute average correlation. In addition, we estimated the Mean-Squared Error (MSE) of the estimator $\hat{S}_j(t)$ based on the R = 500 simulation replicates for each brain location j and time point t. The Total MSE (TMSE) was computed by adding all the MSE's over brain locations and time points. This was done separately for locations in active and inactive regions.

In our simulation studies, the ACS-ICM algorithm had four tuning parameters. The first denoted as $q_o \in (0, 1)$ controlled the degree of stochasticity, with larger values corresponding to less stochasticity and thus less random exploration of the parameter space. When a solution is chosen, another tuning parameter τ_0 controlled the amount of pheromone reinforcing this solution in the information available to the other ants. A third tuning parameter ρ controlled the evaporation of pheromone, and finally a fourth tuning parameter N_{ants} controlled the number of ants. The number of ants (N_{ants}) was fixed at 10, a value for which we have seen generally good performance. This was chosen based on computing efficiency and similar results (objective function values) from using $N_{ants} \geq 10$.

The remaining optimal tuning parameters (q_o, τ_0, ρ) for all simulations cases were chosen using an outer level optimization using the Nelder–Mead algorithm.

3.2. Simulation Results

3.2.1. Evaluation of Neural Source Estimation

We present the average correlation between the estimated values and the truth for the algorithms considered in our study in Appendix A, Table A1. Inspecting Table A1, we observe that for all cases considered for the true number of latent states (either two, three, four, or nine), the estimates obtained from the ACS-ICM algorithm yielded a higher average correlation than those obtained from ICM. In addition, with respect to the number of clusters, ACS-ICM resulted in a higher average correlation than ICM uniformly for all cluster sizes (250, 500, 1000). In summary, the average correlation was significantly improved when estimates were computed using the ACS-ICM algorithm for both large and small numbers of latent states as well as cluster sizes. In addition, we present in Appendix A, Figure A4, violin plots comparing the correlation values obtained from each of the algorithms for different simulation cases across all replicates. These plots show the entire distribution and provide a better assessment of each algorithm for simulation replicates. Observing Figure A4, we can see that ACS-ICM provides the highest correlation values uniformly in all simulation scenarios.

The TMSE for all simulation scenarios is presented in Appendix A, Table A2. To improve the readability of the results from TMSE values, we computed the relative percentage improvement in TMSE of the neural source estimators from ICM to ACS-ICM. Here, using ICM as the reference algorithm, the relative percentage improvement is defined as the ratio of the difference in TMSE between ICM and ACS-ICM to its ICM TMSE value multiplied by 100. The results of this computation are presented in Table 1. In all simulation scenarios for Table 1, ACS-ICM performed better and showed a significant improvement as compared to ICM. Specifically with respect to the number of clusters, ACS-ICM was roughly 10% better than ICM with respect to TMSE when the cluster size was 250. For both small and large numbers of latent states, ACS-ICM was better than ICM in the active region with significant improvements. This shows that ACS-ICM outperforms ICM in active regions using both small and large numbers of latent states. The total MSEs were decomposed into total variance and total squared bias for the same distinct cases of the simulation depicted in Table 1. From the results, when we considered active regions with different numbers of clusters, and observed that ACS-ICM was better than ICM based on the total squared bias due to the percentage of relative change. Based on the total variance we also noticed a similar positive change from ICM to ACS-ICM uniformly for all values of K. It is also clear that for inactive regions, ACS-ICM was better than ICM for both total variance and squared bias for all simulation cases considered. Overall, these results from the TMSE demonstrate a significant improvement obtained from our algorithm when considering total squared bias and variance for our simulation studies. This improvement was observed uniformly across all conditions.

We present in Figure 1 boxplots comparing the final objective function values obtained from each of the algorithms for the different simulation scenarios across all replicates. Again, a clear pattern emerged showing that ACS-ICM yielded the highest objective function values uniformly in all cases. Overall, ACS-ICM outperformed the ICM algorithm uniformly with respect to both neural source estimates and the values of the objective function. This indicates the superiority of the ACS-ICM algorithm over ICM for computing neural source estimates for the spatiotemporal model.

3.2.2. Evaluation of Mixture Component Estimation

In addition to evaluating point estimation and objective function maximization, we also evaluated model selection, comparing $\hat{K}_{ACS}$ and $\hat{K}_{ICM}$, that is, the estimators obtained from ACS-ICM and ICM, respectively. We focused on estimating the number of mixture components and evaluating the sampling distribution of $\hat{K}_{ACS}$ and $\hat{K}_{ICM}$. The following five scenarios were considered in our experiments:

1. Two latent states with Gaussian source activity in the active regions depicted in Appendix A, Figure A1, panel (a).
2. Three latent states with Gaussian source activity in the active regions depicted in Appendix A, Figure A1, panel (c).
3. Four latent states with Gaussian source activity in the active regions depicted in Appendix A, Figure A1, panel (e).
4. Four latent states with Gaussian and sinusoidal source activity in the active regions depicted in Appendix A, Figure A1, panel (g).
5. Nine latent states with Gaussian source activity in the active regions depicted in Appendix A, Figure A2, panel (a).

Table 1. Simulation study I-Percentage of relative improvement in Total Mean-Squared Error (TMSE) of the neural source estimators decomposed into variance and squared bias from ICM to ACS-ICM. This total was obtained separately for locations in active regions and then for the inactive region.

		Active Region			**Inactive Region**		
Algorithm	**Clusters**	**TMSE (%)** %	**(Bias)^2** %	**Variance** %	**TMSE (%)** %	**(Bias)^2** %	**Variance** %
				$K = 2$			
ICM→ACS-ICM	250	9.78	11.11	8.93	9.93	6.15	13.16
ICM→ACS-ICM	500	6.63	4.39	8.57	9.48	9.71	9.26
ICM→ACS-ICM	1000	2.95	2.04	4.03	3.86	1.57	5.70
				$K = 3$			
ICM→ACS-ICM	250	5.10	7.80	2.76	4.97	4.31	5.60
ICM→ACS-ICM	500	24.85	25.42	24.31	15.89	18.26	13.61
ICM→ACS-ICM	1000	36.57	53.61	20	10.83	10	11.61
				$K = 4$			
ICM→ACS-ICM	250	12.24	11.11	12.88	8.90	14.19	4.94
ICM→ACS-ICM	500	17.94	14.75	20.71	2.86	3.10	2.64
ICM→ACS-ICM	1000	29.28	30.30	30.65	2.76	3.62	2.06
				$K = 9$			
ICM→ACS-ICM	250	31.14	22.77	27.65	15.44	13.07	17.58
ICM→ACS-ICM	500	14.83	11.40	18.20	17.0	20.39	13.71
ICM→ACS-ICM	1000	23.79	25.52	22.08	8.65	7.03	10.14

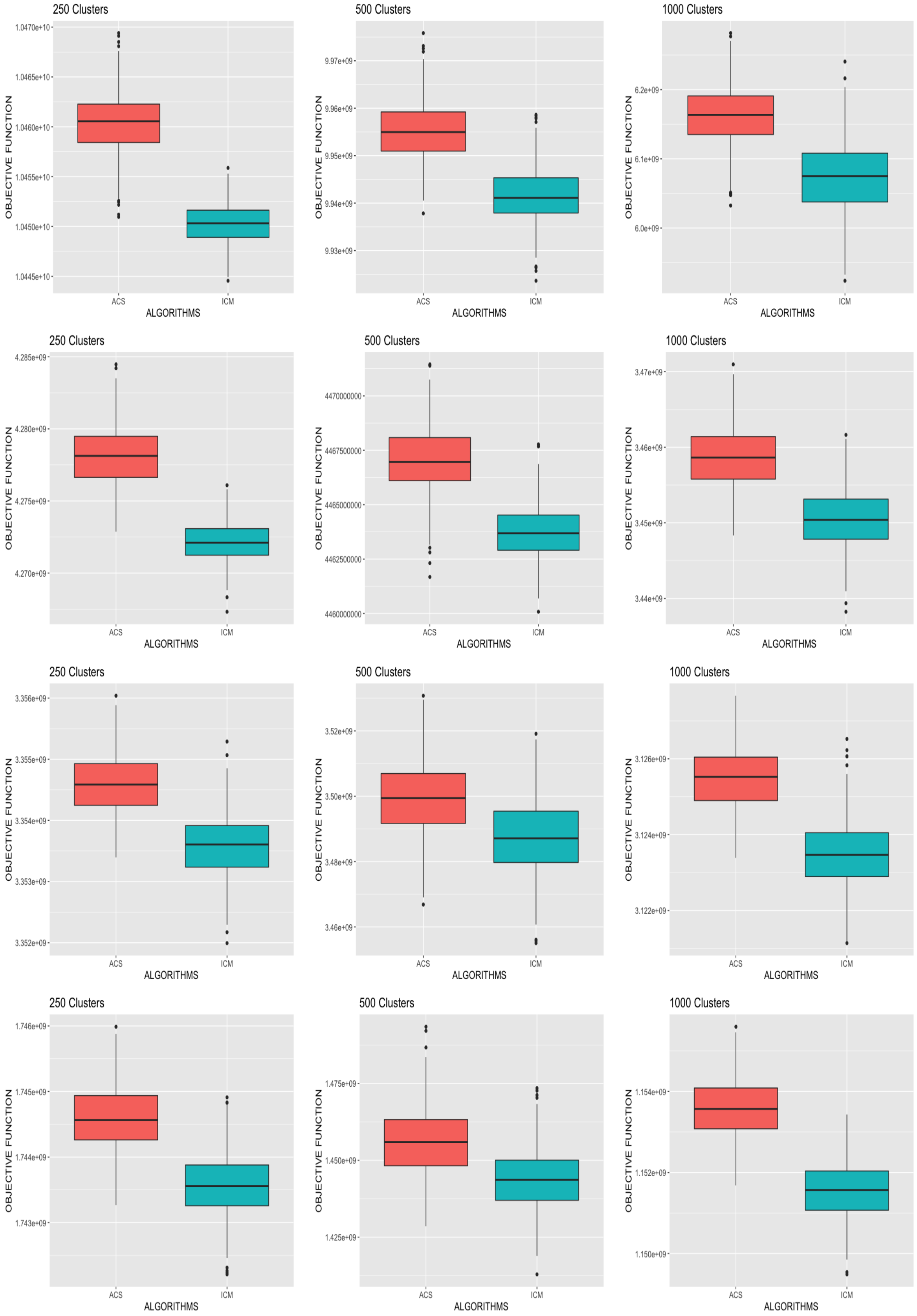


Figure 1. Box-plots comparing the objective function values obtained in the simulation studies for the ICM and ACS-ICM algorithms. The first row corresponds to the case when $K = 2$, second row corresponds to when $K = 3$, third row is when $K = 4$ and the last row is when $K = 9$.

We simulated the data for each of the five scenarios considered, and added 5% Gaussian noise at the sensors with 1000 replicate datasets used in each case. The algorithms were run with an upper bound of $K = 10$ for each of the 5000 simulated datasets. For each dataset, we computed the value of the estimator, and histograms representing the sampling distributions are presented in Figure 2, for each of the five cases above illustrating the sampling distribution of $\hat{K}_{ICM}$ (panels (a)–(e)) and $\hat{K}_{ACS}$ (panels (f)–(j)) corresponding to the first and second row, respectively. Observing Figure 2, where the true signals are well separated in the simulation experiments, in all cases except for the case with a larger number of latent states ($K = 9$), the mode of the sampling distributions corresponds to the true number of latent states for both the ACS-ICM and ICM algorithms. In the case of nine neural sources, ACS-ICM gave better and improved results than ICM. Additionally, Table 2 reports both the bias and mean-squared error of the estimators from ACS-ICM ($\hat{K}_{ACS}$) and ICM ($\hat{K}_{ICM}$). From Table 2, both ACS-ICM and ICM are biased and over-estimated for the small number of latent states but underestimated for the large number of latent states. More importantly, the estimate for the number of mixture components obtained from ACS-ICM exhibited the best performance in terms of both bias and MSE uniformly for all cases considered. This is based on $|\text{Bias}(\hat{K}_{ACS})| < |\text{Bias}(\hat{K}_{ICM})|$ and $\text{MSE}(\hat{K}_{ACS}) < \text{MSE}(\hat{K}_{ICM})$.

We repeated the simulation studies for all five cases but where true signals are less well separated by altering the true signals depicted in Appendix A, Figure A3. We present histograms depicted in Figure 3, for each of the five cases above, illustrating the sampling distribution of $\hat{K}_{ICM}$ (panels (a)–(e)) and $\hat{K}_{ACS}$ (panels (f)–(j)). In this case, the mode of the sampling distribution corresponds to the true number of latent states when $K = 2$ and $K = 3$ but not for the case with four and nine latent states with both algorithms. In Table 2 we compare the bias and mean square error of $\hat{K}_{ICM}$ and $\hat{K}_{ACS}$ under this simulation settings. Similarly, under these settings, ACS-ICM outperformed ICM in terms of the bias and mean square error; thus, $|\text{Bias}(\hat{K}_{ACS})| < |\text{Bias}(\hat{K}_{ICM})|$ and $\text{MSE}(\hat{K}_{ACS}) < \text{MSE}(\hat{K}_{ICM})$. In summary, for model selection, based on the results presented in Table 2, ACS-ICM showed an overall better performance over ICM uniformly for all eight conditions considered.

Whereas the ACS-ICM algorithm showed superiority in terms of quality of source estimates, a drawback is that it is computationally expensive relative to ICM due to its population-based and iterative procedure. Notwithstanding, this might not be a serious challenge for source localization problems, which do not require real-time solutions in most situations. With regards to computation time, on the Niagara cluster running R software on a single core (Intel Skylake 2.4 GHz, AVX512), ICM computed source estimates in approximately 2 min whereas ACS-ICM computed estimates in roughly 6 h and 30 min.

Table 2. Simulation study II—bias and Mean Square Error (MSE) of estimated number of mixture components ($\hat{K}$) from the 1000 simulation replicates when the algorithms were run with $K = 10$.

	$K = 2$		$K = 3$		$K = 4$		$K = 9$	
Algorithm	**Bias($\hat{\mathbf{K}}$)**	**MSE($\hat{\mathbf{K}}$)**	**Bias($\hat{\mathbf{K}}$)**	**MSE($\hat{\mathbf{K}}$)**	**Bias($\hat{\mathbf{K}}$)**	**MSE($\hat{\mathbf{K}}$)**	**Bias($\hat{\mathbf{K}}$)**	**MSE($\hat{\mathbf{K}}$)**
The case where the true signals were well-separated								
ICM	0.11	0.13	0.06	0.42	0.20	0.44	−2.54	6.19
ACS-ICM	0.04	0.06	0.02	0.38	0.10	0.31	−2.01	4.46
The case where the true signals were less well-separated								
ICM	0.11	0.13	0.525	0.58	−1.02	1.63	−4.83	16.12
ACS-ICM	0.05	0.07	0.35	0.41	−1.00	1.31	−3.68	10.47

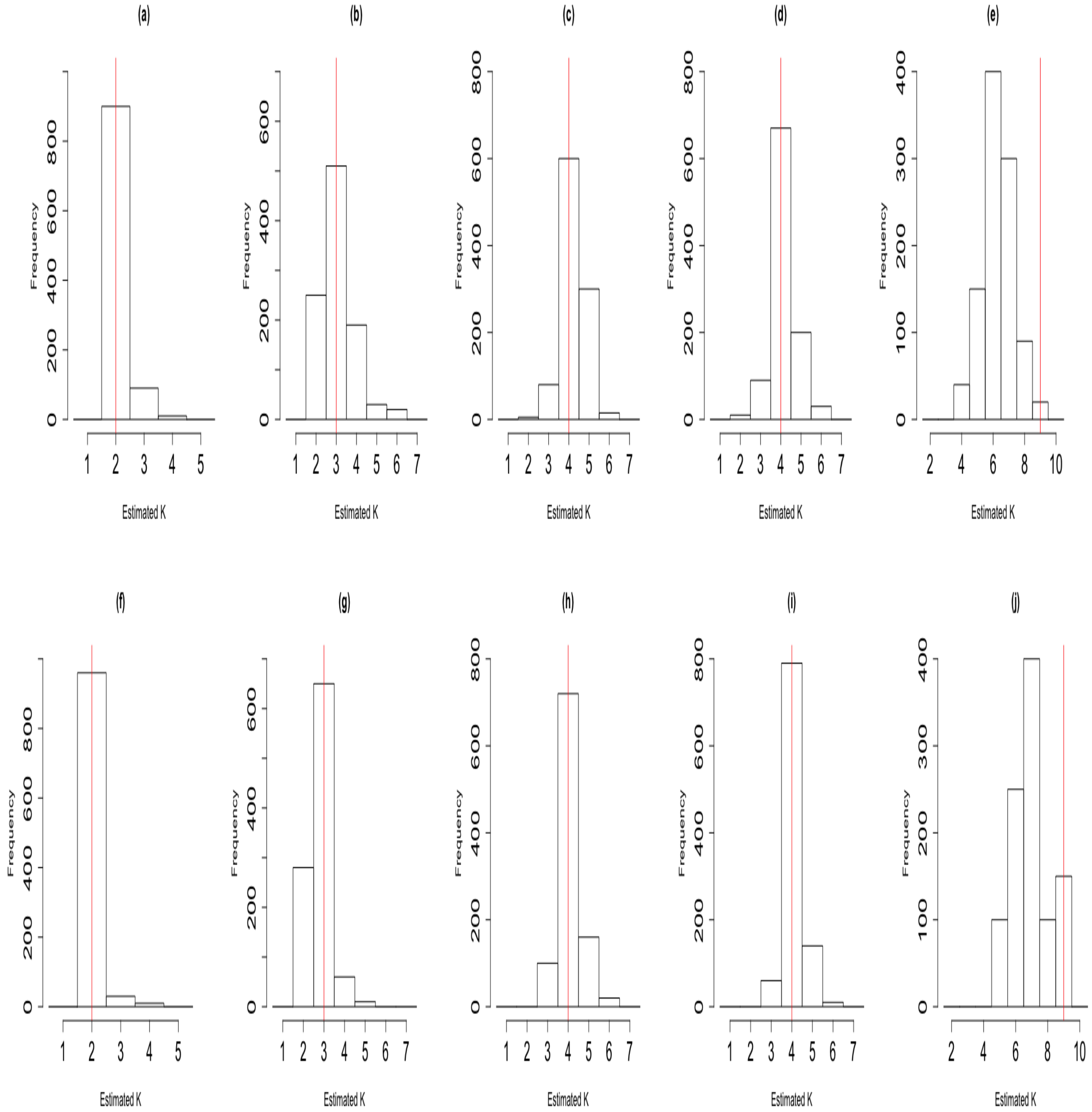


Figure 2. Histograms illustrating the sampling distribution of $\hat{K}$ in the case where the true signals were well separated in the simulation studies. The first row corresponds to the sampling distribution of $\hat{K}_{ICM}$; panel (**a**), $K = 2$; panel (**b**), $K = 3$; panel (**c**), $K = 4$ with three Gaussian signals; panel (**d**), $K = 4$ with two Gaussian signals and one sinusoid; panel (**e**), $K = 9$ with eight Gaussian signals. The second row corresponds to the sampling distribution of $\hat{K}_{ACS}$; panel (**f**), $K = 2$; panel (**g**), $K = 3$; panel (**h**), $K = 4$ with three Gaussian signals; panel (**i**), $K = 4$ with two Gaussian signals and one sinusoid; panel (**j**), $K = 9$ with eight Gaussian signals. In each case the vertical red line indicates the true number of latent states underlying the simulated data.

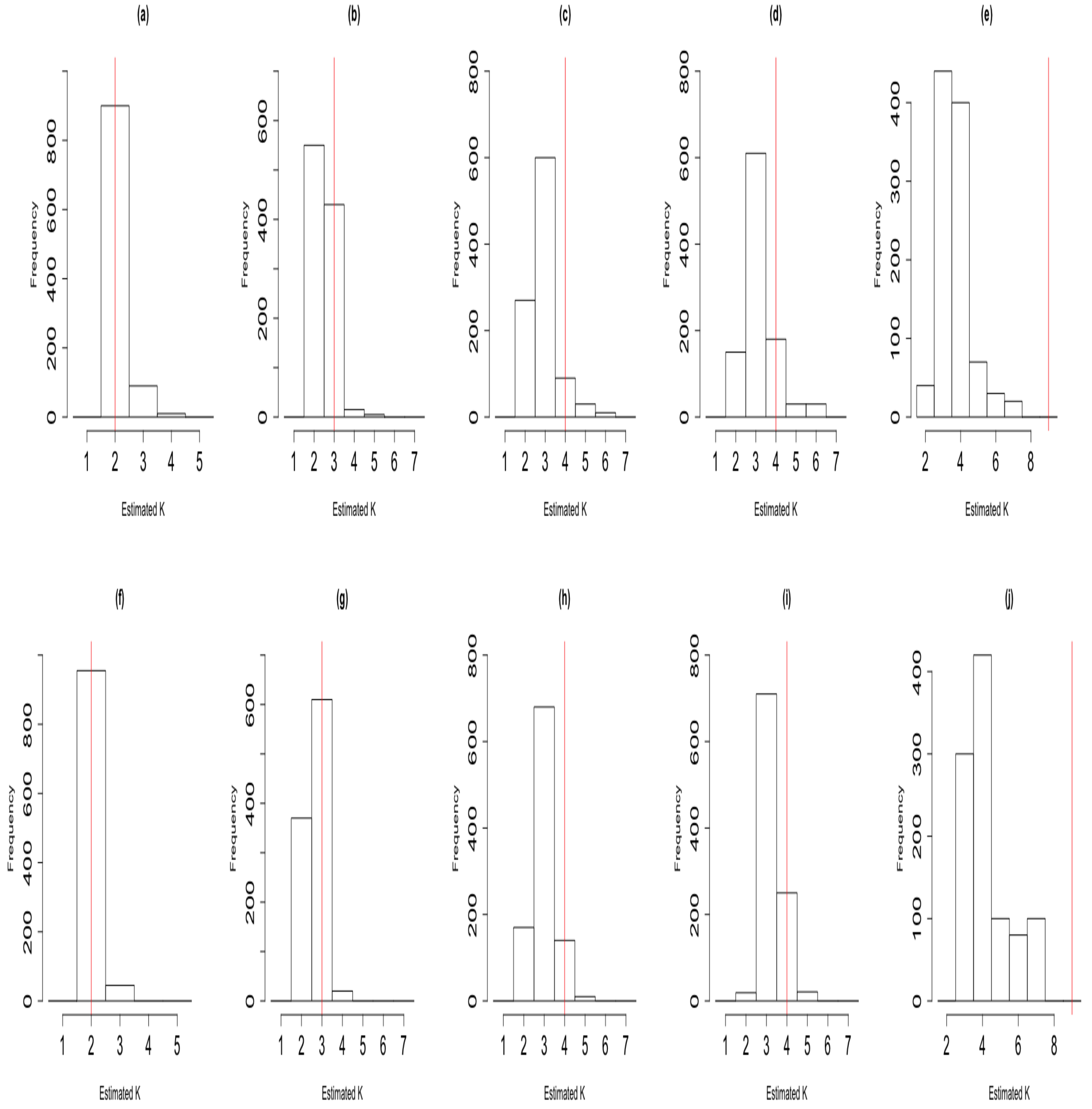


Figure 3. Histograms illustrating the sampling distribution of $\hat{K}$ in the case where the true signals were less well-separated in the simulation studies. The first row corresponds to the sampling distribution of $\hat{K}_{ICM}$; panel (**a**), $K = 2$; panel (**b**), $K = 3$; panel (**c**), $K = 4$ with three Gaussian signals; panel (**d**), $K = 4$ with two Gaussian signals and one sinusoid; panel (**e**), $K = 9$ with eight Gaussian signals. The second row corresponds to the sampling distribution of $\hat{K}_{ACS}$; panel (**f**), $K = 2$; panel (**g**), $K = 3$; panel (**h**), $K = 4$ with three Gaussian signals; panel (**i**), $K = 4$ with two Gaussian signals and one sinusoid; panel (**j**), $K = 9$ with eight Gaussian signals. In each case the vertical red line indicates the true number of latent states underlying the simulated data.

4. Application to Scrambled Face Perception MEG/EEG Data

In this section, we present the application of our methodology for comparison with EEG and MEG data measuring an event-related response to the visual presentation of scrambled faces in a face perception study. In addition, we demonstrate how a nonparametric bootstrap can be used to obtain standard errors, confidence intervals and T-maps. The data from both MEG and EEG were obtained from a single subject in an experimental paradigm that involved repeated random presentation of a picture showing either a face or a scrambled face while the subject was required to make a symmetry judgement. The scrambled faces were created through 2D Fourier transformation, random phase permutation, inverse transformation and outline-masking of each face. The experiment involved a sequence of trials, each lasting 1800 ms, where in each trial the subject was presented with one of the pictures for a period of 600 ms while being required to make a four-way, left–right symmetry judgment while brain activity was recorded over the array. Both scrambled faces and unscrambled faces were presented to the subject; however, our analysis will focus only on trials involving scrambled faces. This produced a multivariate time series for each trial, and the trial-specific time series were then averaged across trials to create a single multivariate time series; the average evoked response is depicted in Figure 4, panel (a), for MEG data, and panel (c), for EEG data. Looking from a spatial perspective, at a given time point, each array recorded a spatial field such as that depicted in Figure 4, panel (b), which shows the MEG spatial field at a particular time point, and Figure 4, panel (d), which shows the EEG spatial field at the same time point. The degree of inter-trial variability was quite low. This experiment was conducted while EEG data were recorded, and then again on the same subject while MEG data were recorded.

The EEG data were acquired on a 128-sensors ActiveTwo system with a high sampling rate of 2048 Hz and down-sampled to 200 Hz. The EEG data were re-referenced to the average over good channels. The resulting EEG data were a trial-specific multivariate time series and contained 128 sensors, 161 time points and 344 trials. For real data analysis, the trial-specific time series were averaged across trials to produce a single average evoked response. The MEG data were acquired on 274 sensors with a CTF/VSM system, with a high sampling rate of 480 Hz and down-sampled to 200 Hz. The MEG data obtained were a trial -specific multivariate time series and contained 274 sensors, 161 time points and 336 trials. We obtained a temporal segment of the data from time point t = 50 to t = 100, resulting in 51 time points for both the EEG and MEG data. The trial-specific time series were averaged across trials to produce a single average evoked response. Detailed description of the data and related analysis can be found in [9,39,40]. In addition, a link to the open access data repository used for analysis can be found here: https://www.fil.ion.ucl.ac.uk/spm/data/mmfaces (accessed on 14 November 2020).

We set the upper bound at $K = 10$ mixture components, voxels as n_v = 560, β = 0.3 (hyperparameter of Potts model) and a cluster size of J = 250. For our real data application, the optimal tuning parameters $(q_o, \tau_0, \rho, N_{ants}) = (0.43, 0.05, 0.64, 10)$ were selected similarly using the Nelder–Mead algorithm. First, the ICM algorithm was run to convergence and the estimates obtained were used as the initial values for the ACS-ICM algorithm. Our primary interest lies in the estimated neural sources $\hat{\mathbf{S}}(t)$ and we computed the total power of these estimated sources obtained from both algorithms at each brain location, which was then mapped onto the cortex. The cortical maps showing the spatial patterns from the estimated power of the reconstructed sources are displayed in Figure 5. The first and second row depict the corresponding results obtained from the ICM and ACS-ICM algorithms, respectively. As shown in Figure 5, the greatest power occurred on the bilateral ventral occipital cortex for both estimated sources from the ACS-ICM and ICM algorithms. Interestingly, the results from ACS-ICM estimates also differed when compared with the results from ICM in the left ventral frontal and right ventral temporal regions. In particular, the ACS-ICM estimate detected higher power, whereas ICM showed low activation in these regions. The estimated source locations of these region is responsible for high-level visual processing. Therefore, the cortical power map seems to represent regions that would be expected to show scrambled face-related activity. To compare the general quality of the estimates from ACS-ICM versus ICM, we show the plot of the final objective function values obtained from the algorithms in Figure 6. We see clearly that the application of ACS-ICM has led to higher quality estimates with much larger posterior density values.

The ACS-ICM algorithm used to maximize the posterior distribution produces only point estimates of the neural source activity. In addition to the point estimates, we applied a nonparametric bootstrap on the trial-specific multivariate time series to obtain confidence interval estimates and characterize the variability in our source estimates, which is another extension to [13]. The interval estimates were constructed by resampling the trial-specific MEG/EEG time series data with replacement. From each resampled dataset, we obtained the average evoked response and then run the ACS-ICM algorithm for a total of 400 nonparametric bootstrap replicates. This procedure was made feasible using parallel computation on a large number of computing cores. We constructed a cortical map of the bootstrap standard deviations of the total power of the estimated source. To account for uncertainty in our point estimates, we constructed a T-map and this is depicted in Figure 7. A T-map is the ratio of the ACS-ICM point estimate of the source activity to its bootstrap standard deviations. The T-map represents the best depiction of reconstructed power since it accounts for standard errors that a simple map of the point estimates does not. Broadly, the T-map seems to indicate similar results to those obtained from point estimates, in particular with respect to high power activation on the bilateral ventral occipital cortex and right ventral temporal region. An interesting observation from the T-map is the detection of a high signal in the left ventral temporal region but a low activation from the point estimate.

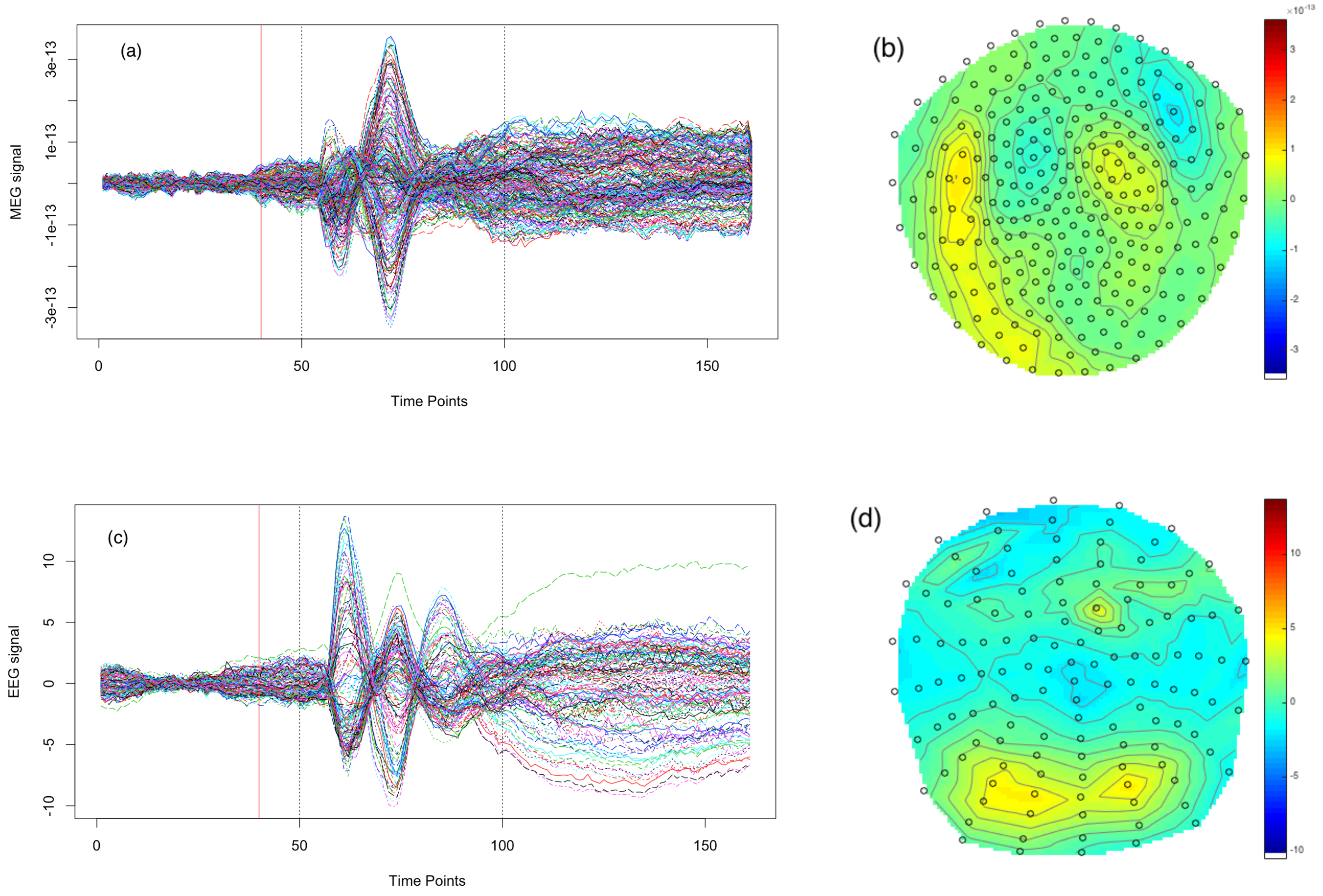


Figure 4. The Magnetoencephalography (MEG) and Electroencephalography (EEG) data considered in the face perception study: panels (**a**,**c**) show the time series observed at each MEG sensor and EEG sensor, respectively; panels (**b**,**d**) depict the spatially interpolated values of the MEG data and the EEG data, respectively, each observed at $t = 80$, roughly 200 ms after presentation of the stimulus. In panels (**b**,**d**) the black circles correspond to the sensor locations after projecting these locations onto a 2-dimensional grid (for presentation).

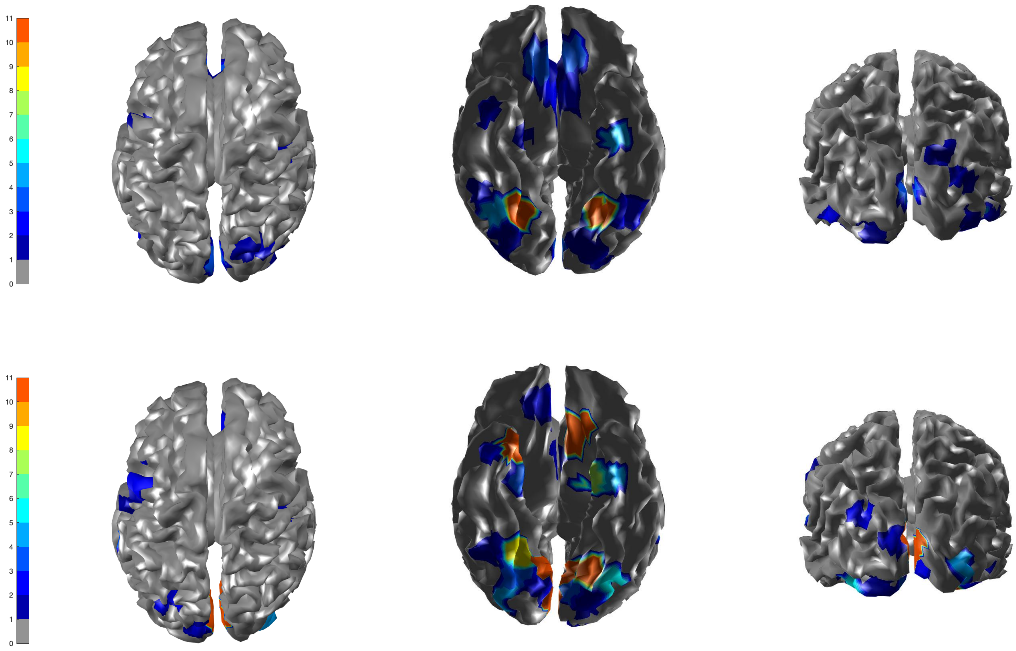


Figure 5. Brain activation for scrambled faces—the power of the estimated source activity $\sum_{t=1}^{T} \hat{S}_j(t)^2$ at each location j of the cortical surface. **Row 1** displays results from our ICM algorithm applied to the combined MEG and EEG data; **Row 2** displays results from ACS-ICM applied to the combined MEG and EEG data.

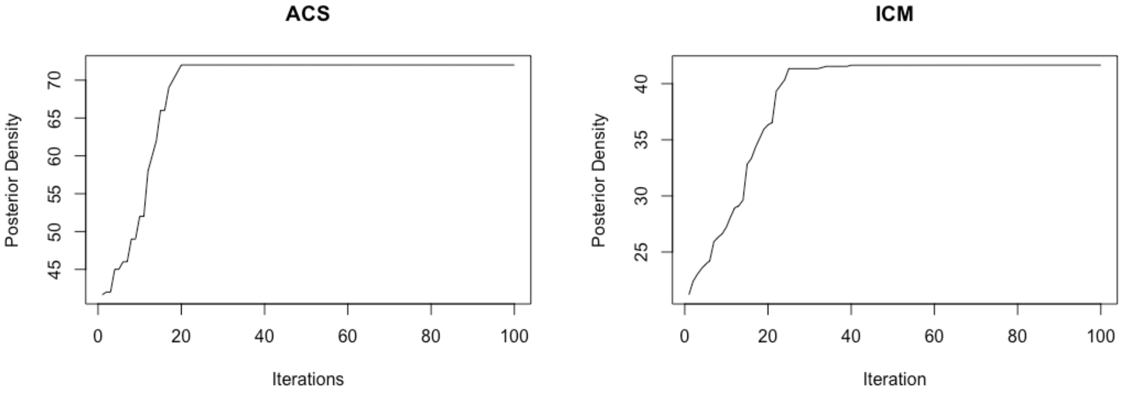


Figure 6. Objective function values obtained from the data with the ACS-ICM (**left**) and ICM (**right**) algorithms.

In addition, we present the temporal summary from our bootstrap replicates representing the interval estimation for the estimated temporal profile of brain activity at the peak location of the T-map. The interval estimate represents a 95% confidence interval depicted in Figure 8. One of the key components of our work is varying the inverse temperature parameter for sensitivity analysis. We fixed the inverse temperature at $\beta = (0.1, 0.44)$ and run the ACS-ICM algorithm to convergence. We run our algorithm together with $K = 10$, n_v = 560 and a cluster size of J = 250. For $\beta = 0.1$, the corresponding results obtained are depicted in the first row of Figure 9. The results indicate activation on the bilateral ventral occipital cortex. Additionally, at $\beta = 0.44$, the power map results from ACS-ICM, depicted in the second row of Figure 9, differ when compared with results from ACS-ICM at $\beta = 0.1$ In particular, the highest power signals occured in the right ventral temporal region where there was low activation for using $\beta = 0.1$.

For our real data application we applied both algorithms with J = 500 clusters so as to evaluate how the performance varies as this tuning parameter changes. The results are displayed in Appendix B. The corresponding results obtained from ACS-ICM are displayed in the second row of Figure A5. Examining Figure A5, ACS-ICM seems to indicate similar results to those obtained from using a tuning parameter of J = 250, in particular with respect to activation on the bilateral ventral occipital cortex. For our sensitivity analysis, we present results obtained from using inverse temperature ($\beta = 0.1$ and $\beta = 0.44$) displayed in Figure A6. We observe that from ACS-ICM, the spatial spread of the high power occurs on the bilateral ventral occipital cortex. In addition, source estimates obtained from ACS-ICM indicate bilateral activation in the occipital cortex, and activation in the right temporal and right frontal regions of the brain. These estimated source locations reveal activation in areas known to be involved in the processing of visual stimuli. More interestingly, ACS-ICM also detected high power in a region on the corpus callosum; given that the inverse problem is ill-posed with an infinite number of possible configurations this may be the reason.

In our real data analysis, the required computation time for ICM was 3 min on a single core (Intel Skylake 2.4 GHz, AVX512) with R software, whereas the computation time for the ACS-ICM was roughly 7 h. The choice of cluster size will have an impact on the computational time required by the algorithm. With regards to ACS-ICM, the required computing time for a cluster size of 250 was approximately 7 h, whereas for a cluster size of 500, ACS-ICM required 12 h of computing time. While there is a substantially increase in computation, the paper has demonstrated uniform improvements in the quality of the solutions, in terms of both source estimation and model selection. Furthermore, the bootstrap can be implemented in parallel on a computing cluster to obtain standard errors with no increase to the required computation time.

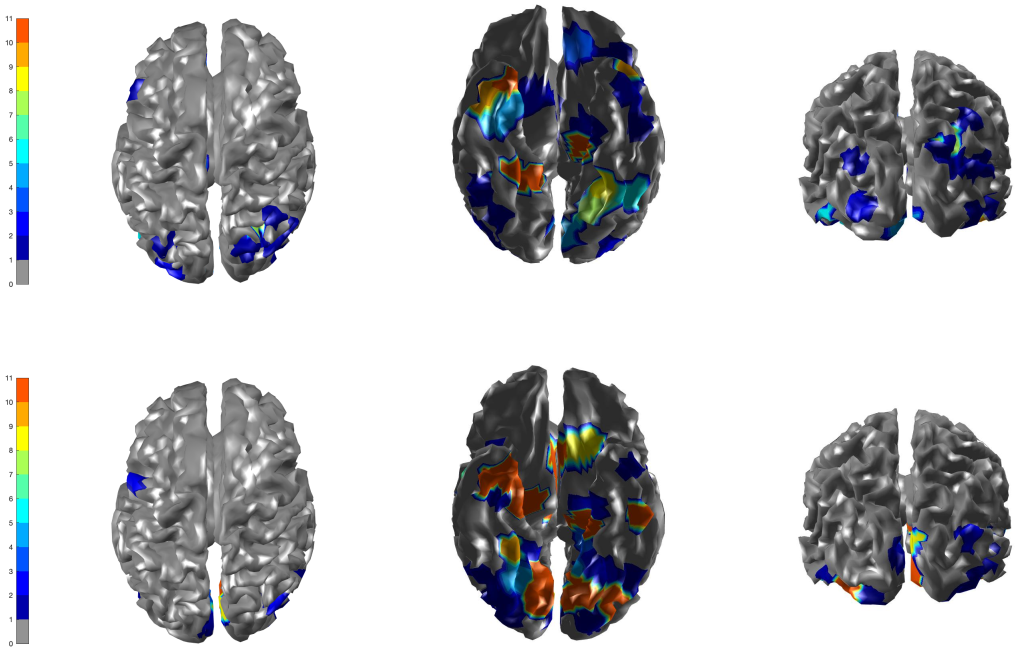


Figure 7. The spatial profile of brain activity from ACS-ICM based on our bootstrap replicates. **Row 1** displays standard deviations of the total power of the estimated source activity; **Row 2** displays the T-map.

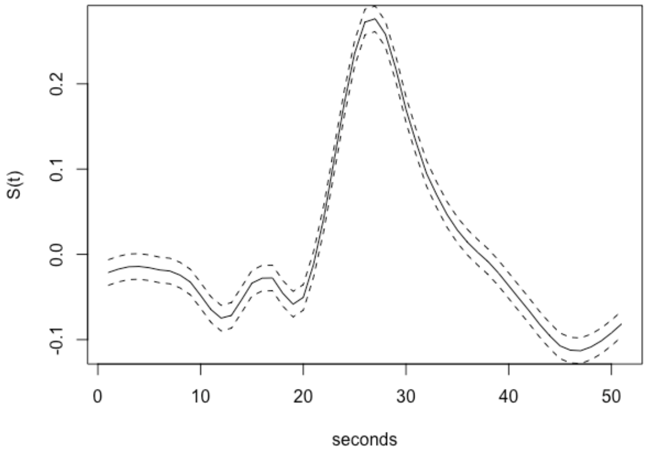


Figure 8. The 95% confidence interval for the estimated temporal profile of brain activity at the peak location of the T-map from the bootstrap replicates.

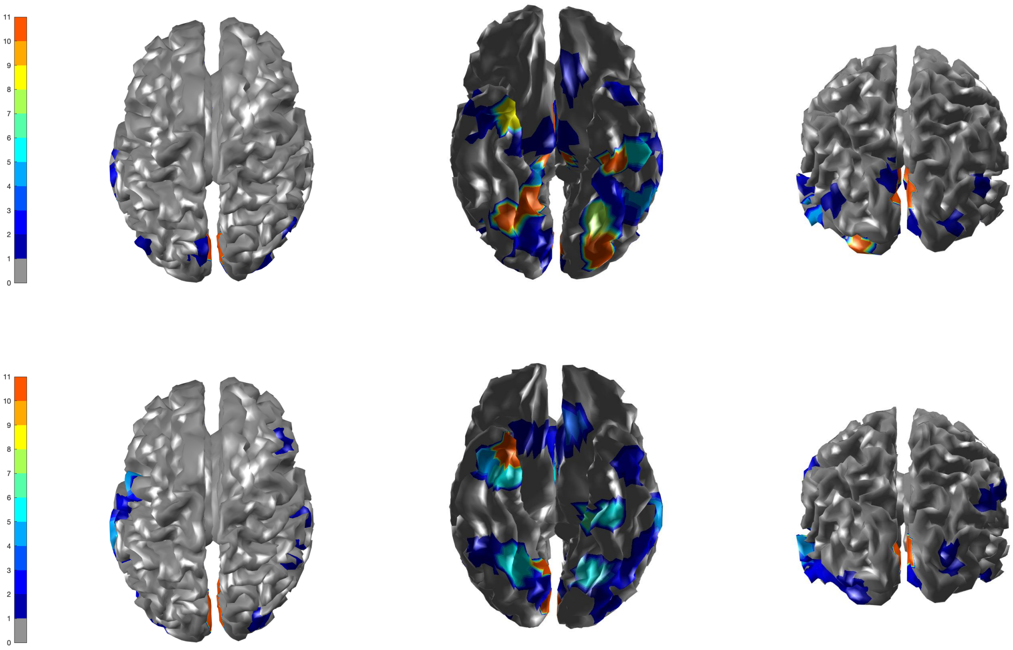


Figure 9. Brain activation for scrambled faces—the power of the estimated source activity $\sum_{t=1}^{T} \hat{S}_j(t)^2$ at each location j of the cortical surface. **Row 1** displays results from our ACS-ICM algorithm applied to the combined MEG and EEG data with $\beta = 0.1$; **Row 2** displays results from ACS-ICM applied to the combined MEG and EEG data with $\beta = 0.44$.

Residual Diagnostics for the Scrambled Faces MEG and EEG Data

We assessed the goodness of fit of the model by checking the residual time series plot, normal quantile–quantile plot and residuals versus fitted values after running the ACS-ICM and ICM algorithms. This was done by computing the residuals for both EEG and MEG after applying both algorithms. The residuals were computed as $\hat{\boldsymbol{\epsilon}}_M(t) = \boldsymbol{M}(t) - \boldsymbol{X}_M\hat{\boldsymbol{S}}(t)$ and $\hat{\boldsymbol{\epsilon}}_E(t) = \boldsymbol{E}(t) - \boldsymbol{X}_E\hat{\boldsymbol{S}}(t)$ at each time point $t = 1, \ldots, T$. The assumption made for the residuals was that they should be draws from a mean-zero Gaussian distribution if the assumed model generated the observed data. The residual time series plot for EEG and MEG from the ACS-ICM algorithm is displayed in Figure 10, panels (a) and (b). The plots from Figure 11, panels (a) and (b), also depicts residuals time series plots obtained from

ICM for EEG and MEG, respectively. Examining the plots, the residual time series plots obtained from both algorithms exhibit similar patterns for MEG and EEG. However, there are significant improvements seen in estimates from ACS-ICM. Specifically for the EEG data, there are sensors with relatively large peaks remaining from the ICM but significant improvements from ACS-ICM as we observe no fewer residuals patterns relative to ICM. In the case of MEG data, we observe that the residuals obtained from ACS-ICM reveal few sensors with peaks remaining as compared to ICM, where there are more sensors with large peaks and residuals.

In Figures 10 and 11, panels (c) and (d), we show plots of the residuals versus fitted values from ACS-ICM and ICM. For the EEG data, the ACS-ICM residuals reveal fewer extreme values with smaller residual patterns but more outliers are seen in the residuals obtained from ICM comparably. The residuals obtained from ICM are characterized by higher values to the left of zero and lower values to the right of zero. In the case of MEG data, the residuals obtained from ACS-ICM also show fewer extreme values with a smaller residual pattern but a similar resemblance for residuals obtained from the ICM algorithm with few extreme values outside the zero band. We observe more extreme values in the residual plot obtained from ICM than that obtained from ACS-ICM. This signifies improvements of the ACS-ICM algorithm over ICM. Inspecting Figure 10, panels (e) and (f), reveals normal quantile–quantile plots for the EEG and MEG residuals obtained from the ACS-ICM algorithm. There is no deviation from normality observed from the EEG and MEG data. Hence, the Gaussian assumption holds from using the ACS-ICM algorithm. In the case of the ICM, in Figure 11, panels (e) and (f) depict the normal quantile–quantile plots for the EEG and MEG data. In this case we observe a clear divergence from the normal distribution for the EEG and MEG residuals. In particular, we see a strong deviation from normality in the left and right tail of the distribution for the EEG data. There is also a deviation from normality in the right tail of the distribution for the MEG data.

In summary, the residual analysis revealed the use of the ACS-ICM algorithm resulted in estimates with a better fit of the spatial mixture model for the EEG and MEG data relative to ICM. Thus our proposed approach leads to improvements in point estimation and model selection uniformly in all settings in simulation studies and in our application with larger objective function values and improved model fit based on residual analysis.

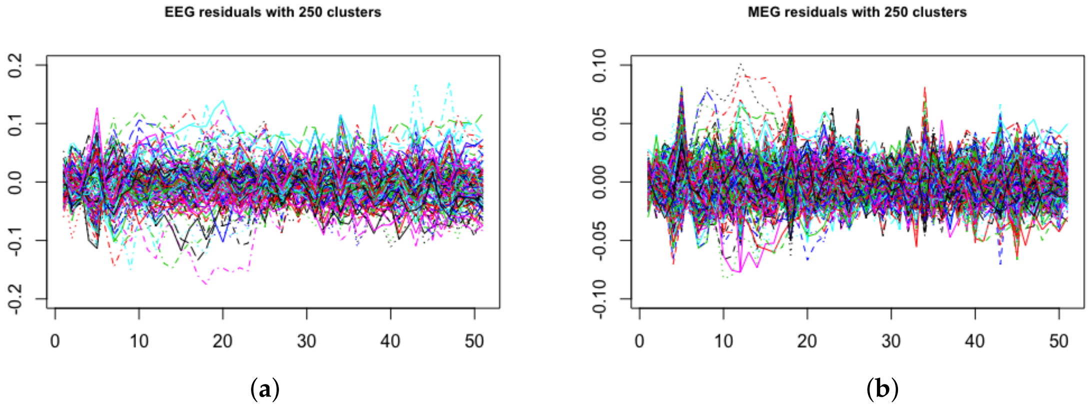
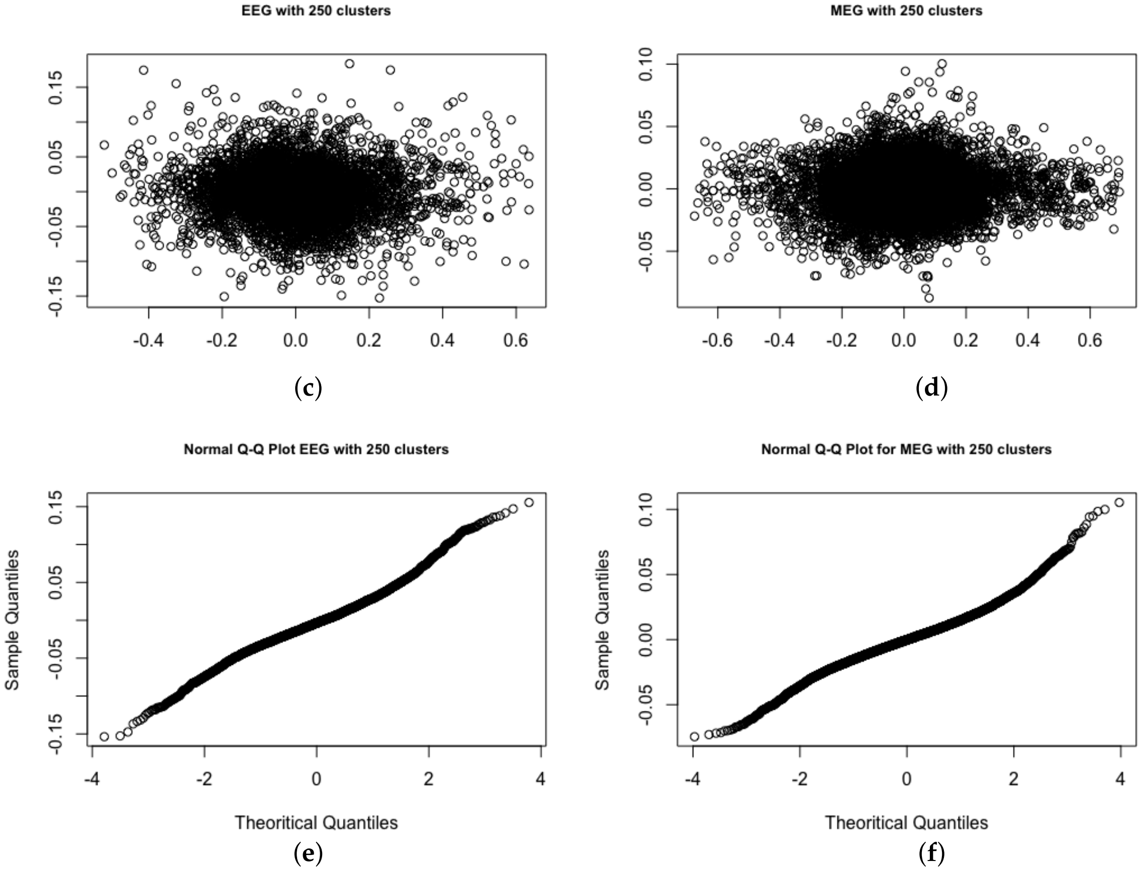


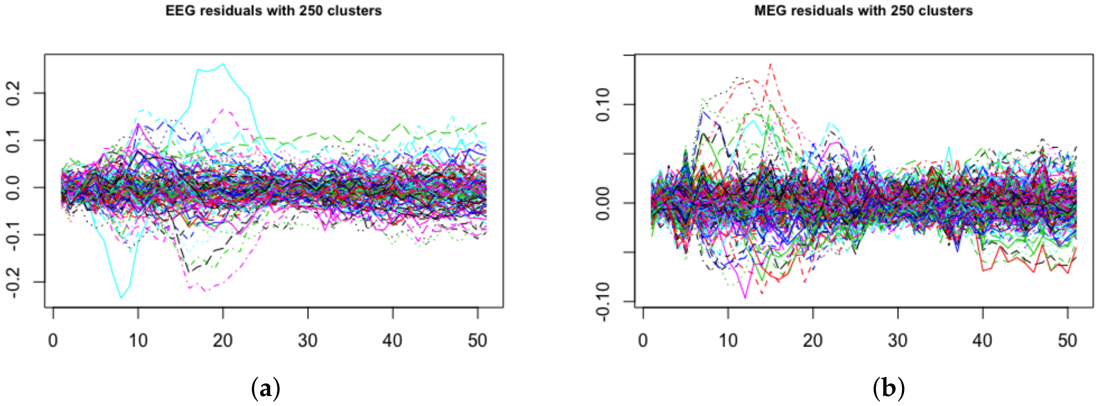
Figure 10. *Cont.*



(**c**) (**d**)

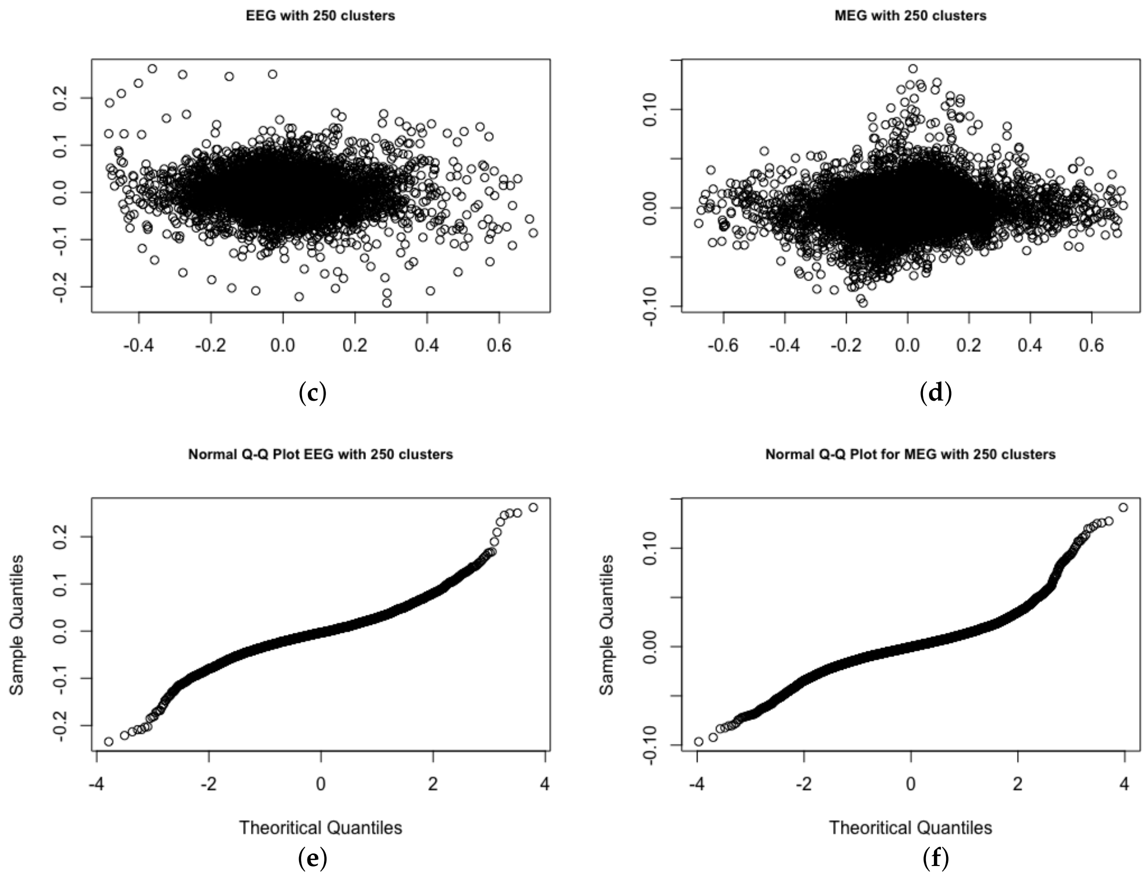
(**e**) (**f**)

Figure 10. Brain activation for scrambled faces using the ACS-ICM algorithm—residual diagnostics: time series of residuals, (**a**) EEG, (**b**) MEG; residuals versus fitted values, (**c**) EEG, (**d**) MEG; residual normal quantile–quantile plots, (**e**) EEG, (**f**) MEG.



(**a**) (**b**)

Figure 11. *Cont.*



(**c**) (**d**)

(**e**) (**f**)

Figure 11. Brain activation for scrambled faces using icm algorithm—residual diagnostics: time series of residuals, (**a**) EEG, (**b**) MEG; residuals versus fitted values, (**c**) EEG, (**d**) MEG; residual normal quantile–quantile plots, (**e**) EEG, (**f**) MEG.

5. Discussion and Conclusions

In this section, we provide numerical results obtained in the data analysis, limitations of the proposed approach and the prospects for future research. We have developed an ACS-ICM algorithm for spatiotemporal modeling of combined MEG/EEG data for solving the neuroelectromagnetic inverse problem. Adopting a Bayesian finite mixture model with a Potts model as a spatial prior, the focus of our work has been to improve source localization estimates, model selection and model fit. The primary contribution is the design and implementation of the ACS-ICM algorithm as an approach for source localization that result in better performance over ICM, which is very positive uniformly in every setting on simulation studies and real data application. Another key development is the technique implemented in choosing the tuning parameters for the ACS-ICM by using an outer level optimization that numerically optimizes the choice of the tuning parameters for this algorithm. This strategy ensures that the optimal tuning parameters based on the data and problem complexity are selected.

5.1. Numerical Results

In our simulation studies, we observed four significant improvements associated with ACS-ICM over ICM: (1) ACS-ICM neural source estimates provided improved correlation between estimated and truth sources uniformly across all settings considered; (2) the objective function values obtained from the posterior density values for ACS-ICM were larger than those obtained from ICM uniformly across all settings considered; (3) ACS-ICM showed significant improvement with respect to the total mean square error for all cluster sizes considered compared to ICM; (4) ACS-ICM exhibited improved performance in terms of both bias and mean square error for the non-regular problem of estimating number of mixture components. Moreover, the application of ACS-ICM to real data led to higher quality estimates with larger maximized posterior density values. These improvements have demonstrated the advantage of the ACS-ICM algorithm when compared with ICM in both the face perception analysis as well as the simulation studies. In addition to implementing the ACS-ICM algorithm for point estimation, we demonstrated how a nonparametric bootstrap can be used to obtain standard errors, confidence intervals and T-maps for the proposed methodology. This was done to account for uncertainty in our point estimates of the neural source activity.

5.2. Limitations of the Proposed Approach

An important limitation of the simulation studies is the use of white noise added to the signals. This is because MEG/EEG data would have structured noise that arise from, e.g., motion, and such noise would be spatially correlated. The spatially correlated noise will make the simulation scenarios more challenging, which we expect to result in a decline in performance. We did not pursue this scenario in our simulation and we will consider it in our future studies.

5.3. Prospects for Future Research

In our current work, we are implementing ACS-ICM for the spatial mixture model developed in [13]. We hope in the future to extend the model by considering a robust error structure in the MEG/EEG model. The model currently assumes that the errors are independent in time. This will be extended by allowing for an autoregressive structure. A second extension would be to relax the assumption that the errors have a Gaussian distribution by incorporating a multivariate t distribution for the error terms. Integrating these extensions, we will develop a new joint model for the MEG/EEG data and implement the ACS-ICM and ICM algorithms for the neuroelectromagnetic inverse problem.

Furthermore, when we obtained the source estimates from the ACS-ICM algorithm, we mapped a function of them (the total power) on the cortex and in that map we used no thresholding. That is to say, the locations were not thresholded so we can see all the locations with estimated power. For our future studies we hope to map the total power on the cortex with a threshold so that we can see the locations with highest power. In a better way to choose the threshold, our next objective is to extend this work by implementing thresholding of cortical maps using random field theory [41]. Random field theory is mainly applied in dealing with thresholding problems encountered in functional imaging. This is used to solve the problem of finding the height threshold for a smooth statistical map, which gives the required family-wise error rate. In going forward with our current work, the idea is to take the point estimate obtained from ACS-ICM and standard errors (obtained from bootstrap) to provide estimates of p-values for t-statistics pertaining to the number of activated voxels comprising a particular region.

It should be noted that the ACS-ICM algorithm and spatial model developed can also be applied to studies involving multiple subjects. Expanding from a single subject model to a model developed for multiple subjects would be of great interest for the MEG/EEG inverse problem. This will be based on developing a fully Bayesian analysis based on a divide and conquer Markov Chain Monte Carlo (MCMC) method [42]. This approach for Bayesian computation with multiple subjects is to partition the data into partitions, perform local inference for each piece separately, and combine the results to obtain a global posterior approximation.

Author Contributions: Conceptualization, E.A.O. and F.S.N.; methodology, E.A.O., F.S.N. and Y.S.; formal analysis, E.A.O.; writing—original draft preparation, E.A.O.; writing—review and editing, E.A.O., S.E.A., Y.S. and F.S.N.; supervision, F.S.N. and S.E.A.; funding acquisition, F.S.N. and S.E.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Sciences and Engineering Research Council of Canada (NSERC) and supported by infrastructure provided by Compute Canada (www.computecanada.ca (accessed on 14 November 2020) and data sourced from [13].

Institutional Review Board Statement: Not applicable.

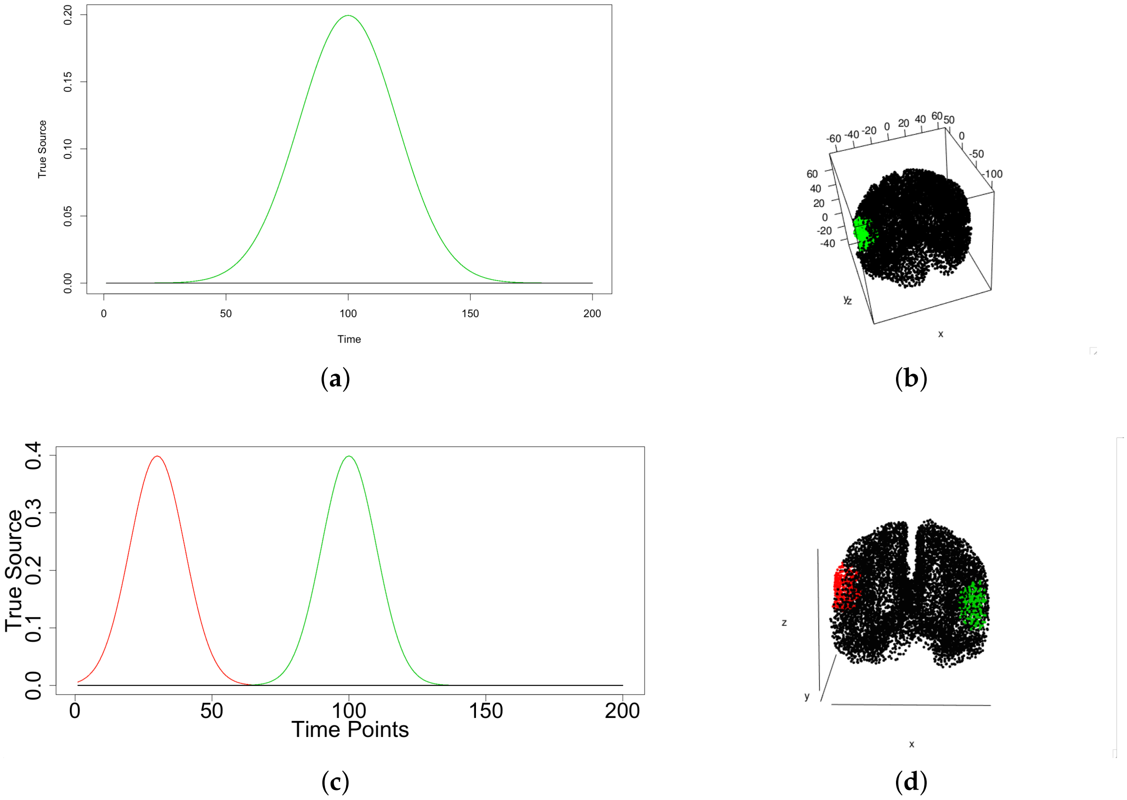
Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here https://www.fil.ion.ucl.ac.uk/spm/data/mmfaces (accessed on 26 January 2021).

Acknowledgments: This research is supported by the Visual and Automated Disease Analytics (VADA) graduate training program.

Conflicts of Interest: The authors declare no conflict of interest.

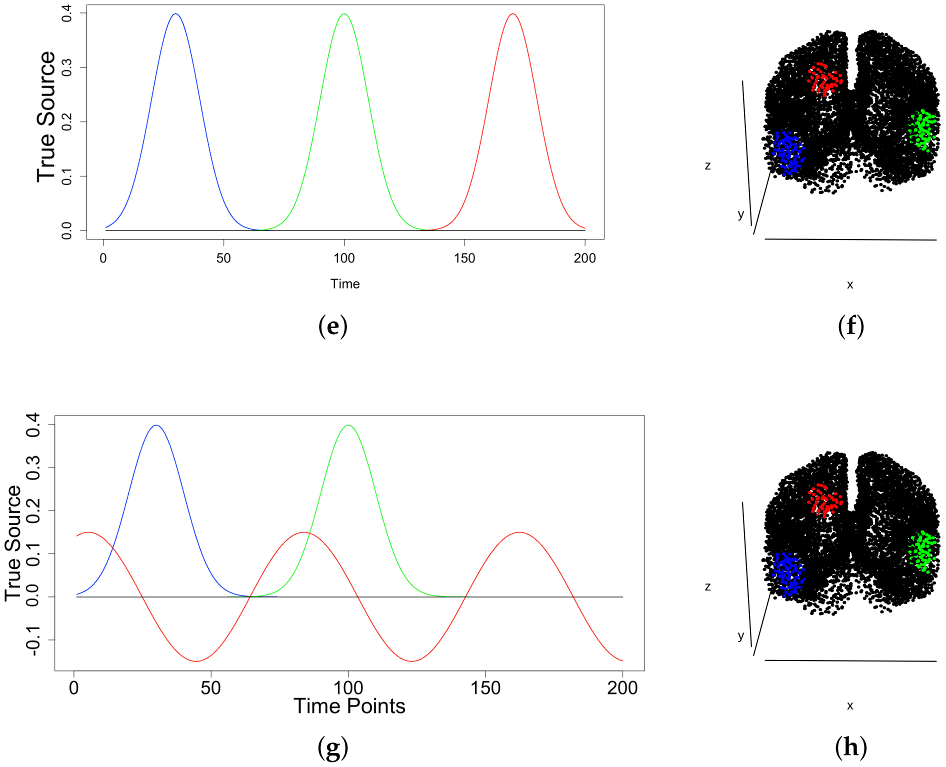
Appendix A



(**a**) (**b**)

(**c**) (**d**)

Figure A1. *Cont.*



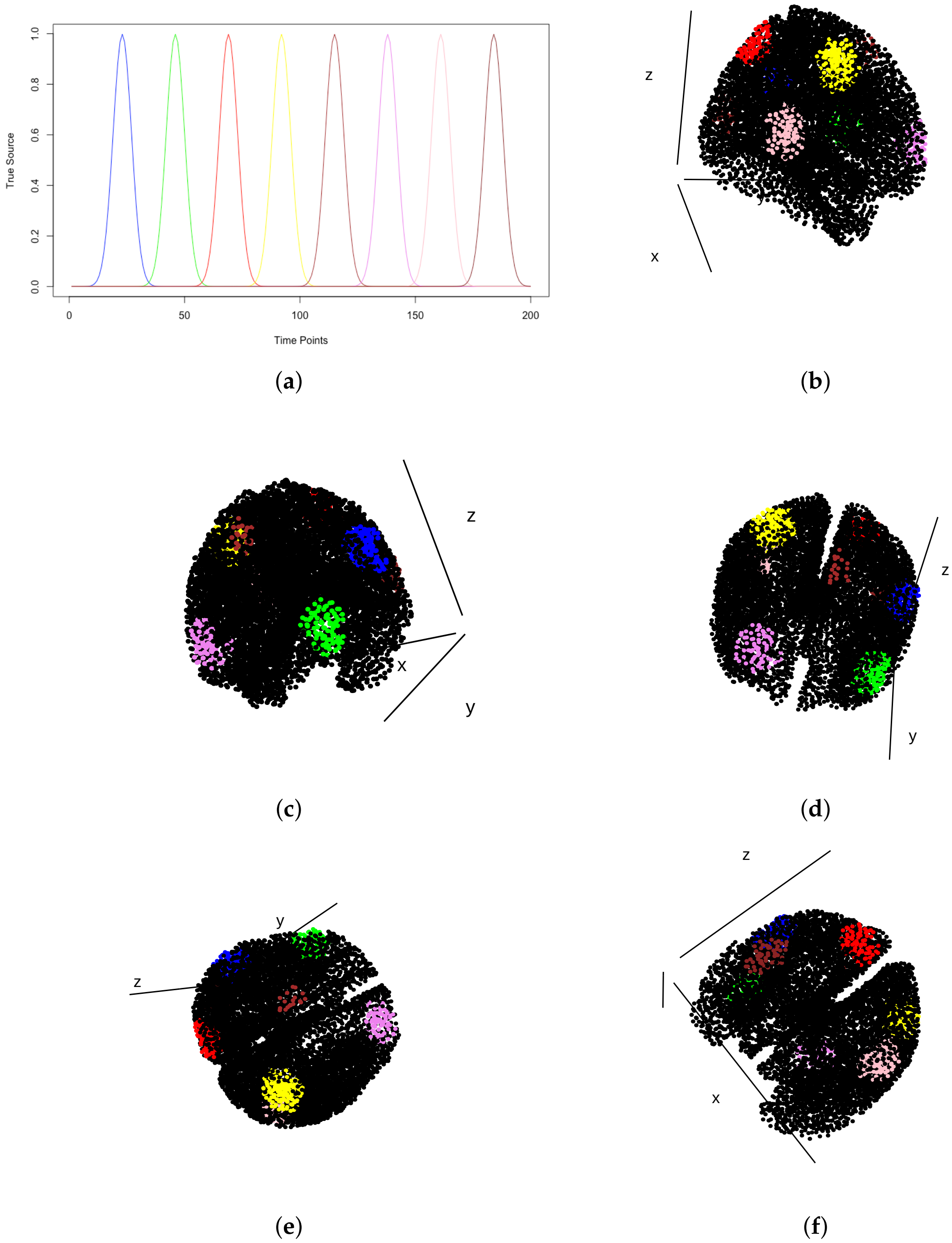
(**e**) (**f**)

(**g**) (**h**)

Figure A1. The true signal $S_j(t)$ used in each of the distinct active and inactive regions in the simulation studies of Section 3 for $K = 2$; panel (**a**), $K = 3$; panel (**c**) and $K = 4$; panel (**e**) & (**g**) are depicted in the left column. The right column presents the true partition of the cortex into active and inactive states for the corresponding states for $K = 2$; panel (**b**), $K = 3$; panel (**d**) and $K = 4$; panel (**f**) & (**h**).

Table A1. Simulation study I—Average (Ave.) correlation between the neural source estimates and the true values for the ICM and ACS-ICM algorithms. The simulation study is based on $R = 500$ simulation replicates where each replicate involves the simulation of MEG and EEG data based on a known configuration of the neural activity depicted in Appendix A, Figures A1 and A2.

		***K* = 2**	***K* = 3**	***K* = 4**	***K* = 9**
Algorithm	**Clusters**	**Ave. Corr.**	**Ave. Corr.**	**Ave. Corr.**	**Ave. Corr.**
ICM	250	0.60	0.63	0.62	0.54
ACS-ICM	250	0.64	0.67	0.63	0.59
ICM	500	0.53	0.55	0.49	0.44
ACS-ICM	500	0.56	0.61	0.53	0.46
ICM	1000	0.41	0.43	0.40	0.37
ACS-ICM	1000	0.46	0.47	0.45	0.43

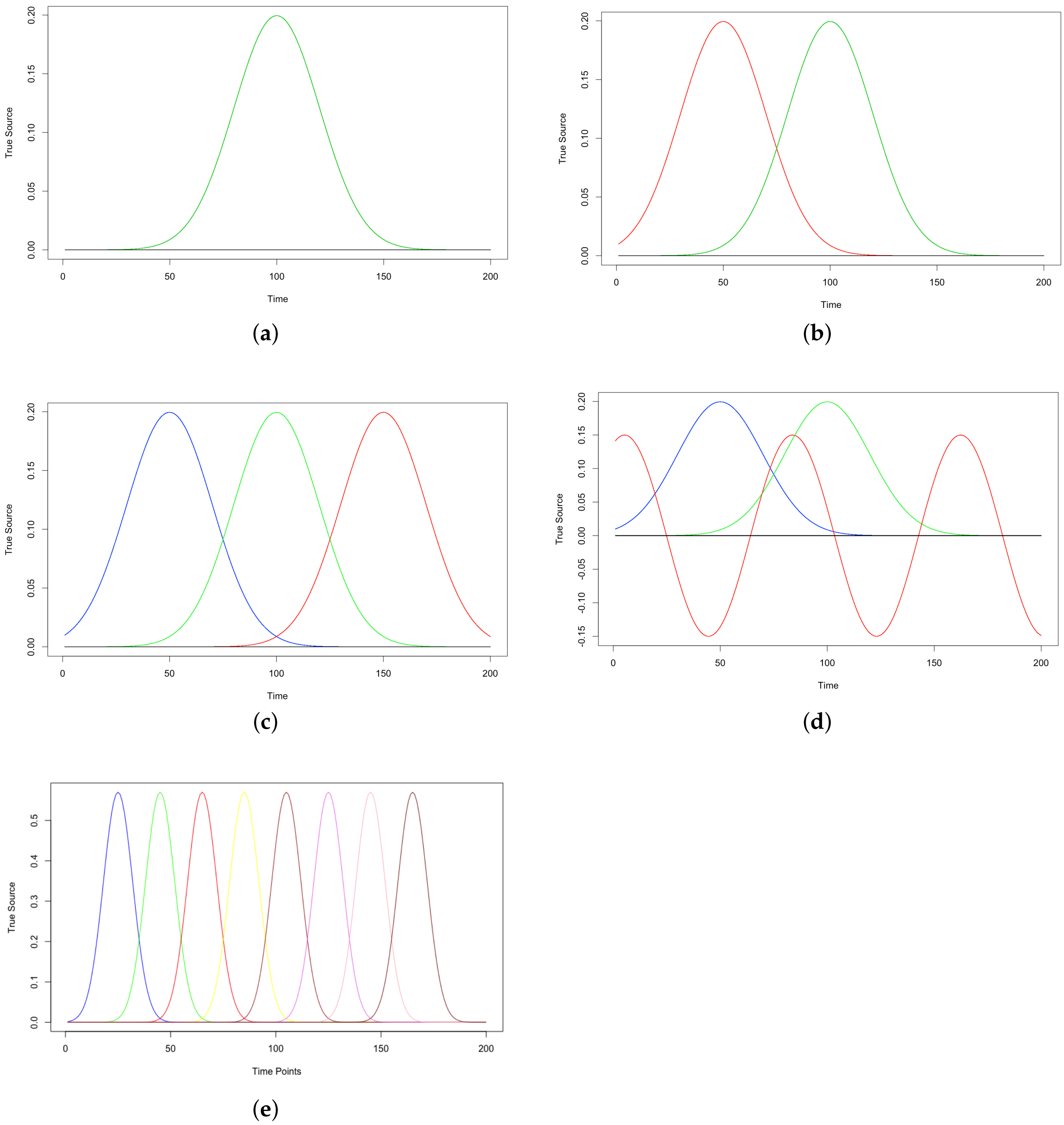


(**a**) (**b**)

(**c**) (**d**)

(**e**) (**f**)

Figure A2. The true signal $S_j(t)$ in panel (**a**) and true partition of the cortex into active and inactive states for the case of $K = 9$ states (panel **b**–**f**) used in simulation studies of Section 3.



(**a**)

(**b**)

(**c**)

(**d**)

(**e**)

Figure A3. The true signal $S_j(t)$ used in in each of the distinct active and inactive regions for $K = 2$; panel (**a**), $K = 3$; panel (**b**), $K = 4$; panel (**c**) & (**d**), $K = 9$; panel (**e**) in the simulation study of Section 3.2, in the second part of the study where the mixture components were less well separated.

Table A2. Simulation study I—Total Mean-Squared Error (TMSE) of the neural source estimators decomposed into variance and squared bias for the ICM and ACS-ICM algorithms. This total was obtained separately for locations in active regions and then for the inactive region.

		Active Region			**Inactive Region**		
Algorithm	**Clusters**	**TMSE**	**(Bias)^2**	**Variance**	**TMSE**	**(Bias)^2**	**Variance**
				$K = 2$			
ICM	250	92	36	56	141	65	76
ACS-ICM	250	83	32	51	127	61	66
ICM	500	196	91	105	211	103	108
ACS-ICM	500	183	87	96	191	93	98
ICM	1000	271	147	124	285	127	158
ACS-ICM	1000	263	144	119	274	125	149
				$K = 3$			
ICM	250	490	237	253	523	255	268
ACS-ICM	250	465	219	246	497	244	253
ICM	500	1203	582	621	705	345	360
ACS-ICM	500	904	434	470	593	282	311
ICM	1000	1657	817	840	674	321	353
ACS-ICM	1000	1051	379	672	601	289	312
				$K = 4$			
ICM	250	776	378	396	674	289	385
ACS-ICM	250	681	336	345	614	248	366
ICM	500	1404	651	753	804	387	417
ACS-ICM	500	1152	555	597	781	375	406
ICM	1000	2493	1100	1393	796	359	437
ACS-ICM	1000	1763	797	966	774	346	428
				$K = 9$			
ICM	250	2100	918	1182	1541	727	814
ACS-ICM	250	1446	709	737	1303	632	671
ICM	500	2515	1246	1269	1260	618	642
ACS-ICM	500	2142	1104	1038	1046	492	554
ICM	1000	3561	1720	1839	1549	740	809
ACS-ICM	1000	2714	1281	1433	1415	688	727

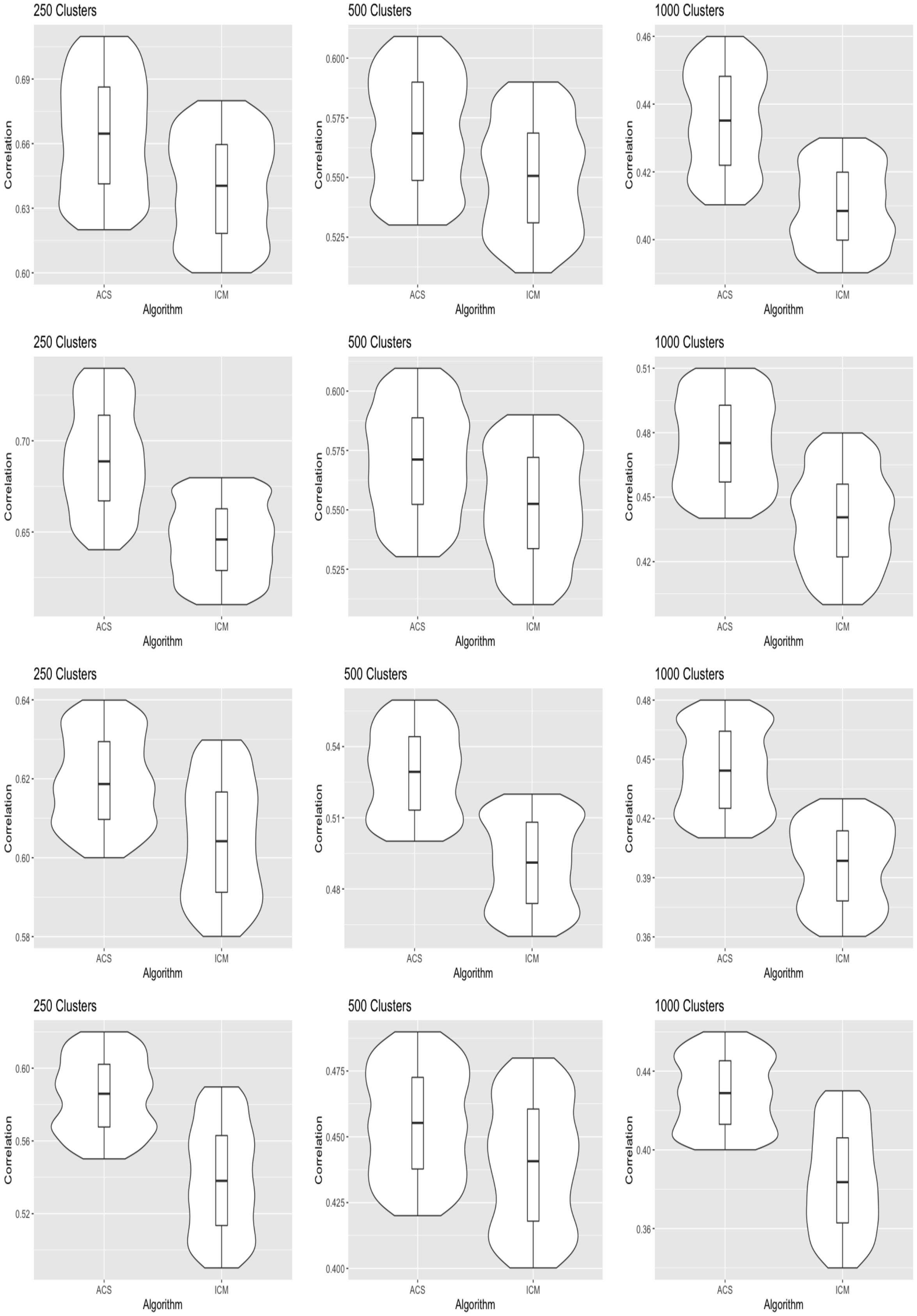


Figure A4. Violin plots comparing the correlation values obtained in the simulation studies for the ICM and ACS-ICM algorithms. The first row corresponds to the case when $K = 2$, the second row corresponds to when $K = 3$, the third row is when $K = 4$ and the last row is when $K = 9$.

Appendix B

We present results from the ACS-ICM and ICM algorithms with tuning parameters for the cluster size of 500. The cortical maps displaying the spatial patterns of the total power for estimated sources are represented below.

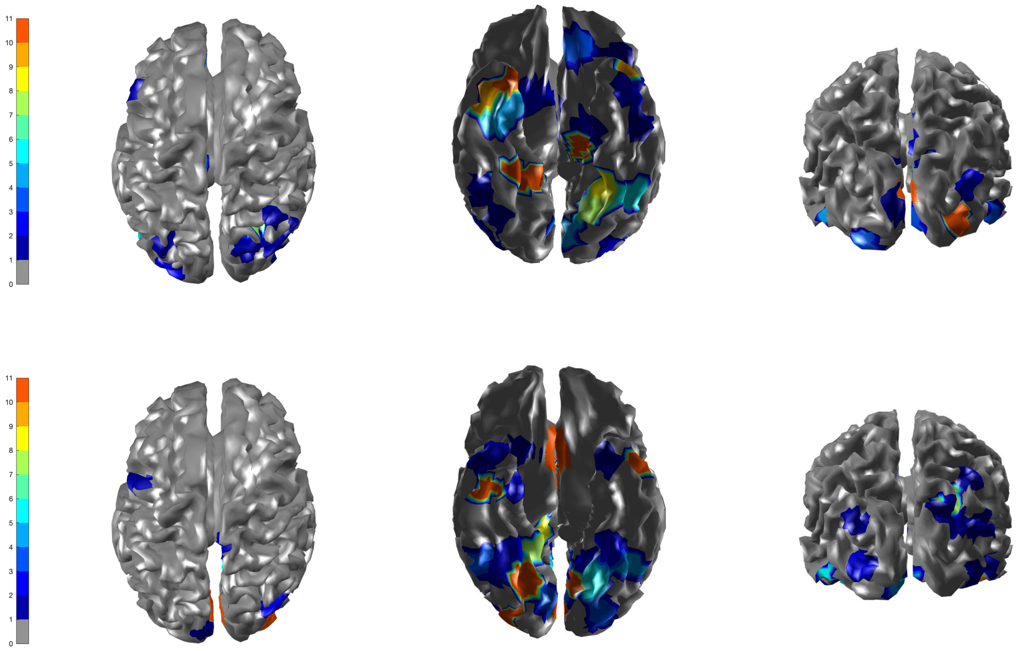


Figure A5. Brain activation for scrambled faces—the power of the estimated source activity $\sum_{t=1}^{T} \hat{S}_j(t)^2$ at each location j of the cortical surface. **Row 1** displays results from our ICM algorithm applied to the combined MEG and EEG data; **Row 2** displays results from ACS applied to the combined MEG and EEG data.

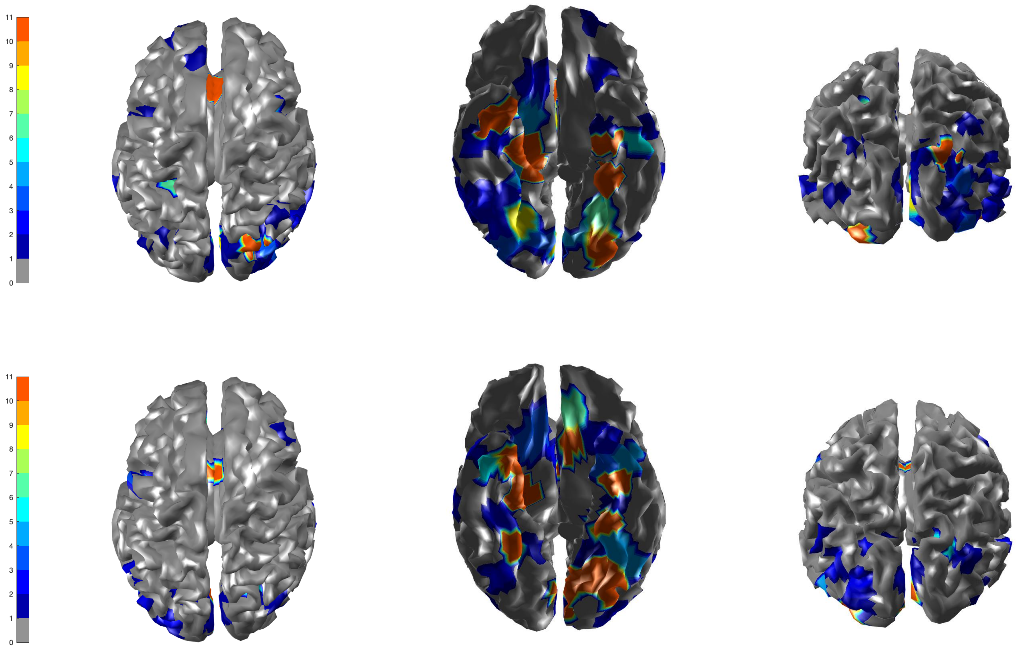


Figure A6. Brain activation for scrambled faces—the power of the estimated source activity $\sum_{t=1}^{T} \hat{S}_j(t)^2$ at each location j of the cortical surface. **Row 1** displays results from our ACS applied to the combined MEG and EEG data with $\beta = 0.1$; **Row 2** displays results from ACS applied to the combined MEG and EEG data with $\beta = 0.44$.

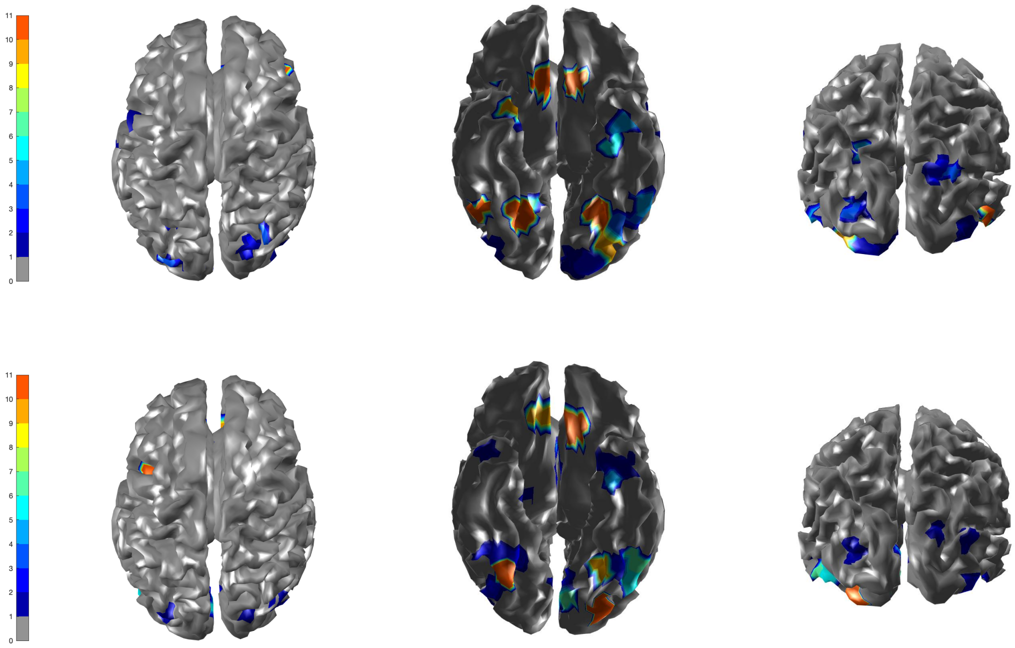


Figure A7. The spatial profile of brain activity from ACS-ICM based on our bootstrap replicates. **Row 1** displays standard deviations of the total power of the estimated source activity; **Row 2** displays the T-map.

References

1. Chowdhury, R.A.; Zerouali, Y.; Hedrich, T.; Heers, M.; Kobayashi, E.; Lina, J.M.; Grova, C. MEG–EEG information fusion and electromagnetic source imaging: From theory to clinical application in epilepsy. *Brain Topogr.* **2015**, *28*, 785–812. [CrossRef]
2. Zhang, X.; Lei, X.; Wu, T.; Jiang, T. A review of EEG and MEG for brainnetome research. *Cogn. Neurodyn.* **2014**, *8*, 87–98. [CrossRef]
3. Baillet, S.; Mosher, J.C.; Leahy, R.M. Electromagnetic brain mapping. *IEEE Signal Process. Mag.* **2001**, *18*, 14–30. [CrossRef]
4. Sorrentino, A.; Piana, M. Inverse Modeling for MEG/EEG data. In *Mathematical and Theoretical Neuroscience*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 239–253.
5. Giraldo-Suarez, E.; Martínez-Vargas, J.D.; Castellanos-Dominguez, G. Reconstruction of neural activity from EEG data using dynamic spatiotemporal constraints. *Int. J. Neural Syst.* **2016**, *26*, 1650026. [CrossRef]
6. Long, C.J.; Purdon, P.L.; Temereanca, S.; Desai, N.U.; Hämäläinen, M.S.; Brown, E.N. State-space solutions to the dynamic magnetoencephalography inverse problem using high performance computing. *Ann. Appl. Stat.* **2011**, *5*, 1207. [CrossRef]
7. Rytsar, R.; Pun, T. EEG source reconstruction using global optimization approaches: Genetic algorithms versus simulated annealing. *Int. J. Tomogr. Simul.* **2010**, *14*, 83–94.
8. Wipf, D.; Nagarajan, S. A unified Bayesian framework for MEG/EEG source imaging. *NeuroImage* **2009**, *44*, 947–966. [CrossRef] [PubMed]
9. Henson, R.N.; Mouchlianitis, E.; Friston, K.J. MEG and EEG data fusion: Simultaneous localisation of face-evoked responses. *Neuroimage* **2009**, *47*, 581–589. [CrossRef] [PubMed]
10. Friston, K.; Harrison, L.; Daunizeau, J.; Kiebel, S.; Phillips, C.; Trujillo-Barreto, N.; Henson, R.; Flandin, G.; Mattout, J. Multiple sparse priors for the M/EEG inverse problem. *NeuroImage* **2008**, *39*, 1104–1120. [CrossRef] [PubMed]
11. Daunizeau, J.; Friston, K.J. A mesostate-space model for EEG and MEG. *NeuroImage* **2007**, *38*, 67–81. [CrossRef]
12. Olier, I.; Trujillo-Barreto, N.J.; El-Deredy, W. A switching multi-scale dynamical network model of EEG/MEG. *Neuroimage* **2013**, *83*, 262–287. [CrossRef] [PubMed]
13. Song, Y.; Nathoo, F.; Babul, A. A Potts-mixture spatiotemporal joint model for combined magnetoencephalography and electroencephalography data. *Can. J. Stat.* **2019**, *47*, 688–711. [CrossRef]
14. Dorigo, M.; Gambardella, L.M. Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Trans. Evol. Comput.* **1997**, *1*, 53–66. [CrossRef]
15. İnkaya, T.; Kayalıgil, S.; Özdemirel, N.E. Ant colony optimization based clustering methodology. *Appl. Soft Comput.* **2015**, *28*, 301–311. [CrossRef]
16. Parpinelli, R.S.; Lopes, H.S.; Freitas, A.A. Data mining with an ant colony optimization algorithm. *IEEE Trans. Evol. Comput.* **2002**, *6*, 321–332. [CrossRef]

17. Sharma, S.; Buddhiraju, K.M. Spatial–spectral ant colony optimization for hyperspectral image classification. *Int. J. Remote Sens.* **2018**, *39*, 2702–2717. [CrossRef]
18. Opoku, E.A.; Ahmed, S.E.; Nelson, T.; Nathoo, F.S. Parameter and Mixture Component Estimation in Spatial Hidden Markov Models: A Comparative Analysis of Computational Methods. In *International Conference on Management Science and Engineering Management*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 340–355.
19. Ouadfel, S.; Batouche, M. Ant colony system with local search for Markov random field image segmentation. In Proceedings of the 2003 International Conference on Image Processing (Cat. No. 03CH37429), Barcelona, Spain, 14–17 September 2003; Volume 1, pp. 1–133.
20. Shweta, K.; Singh, A. An effect and analysis of parameter on ant colony optimization for solving travelling salesman problem. *Int. J. Comput. Sci. Mob. Comput* **2013**, *2*, 222–229.
21. Guan, B.; Zhao, Y.; Li, Y. An Ant Colony Optimization Based on Information Entropy for Constraint Satisfaction Problems. *Entropy* **2019**, *21*, 766. [CrossRef]
22. Lecaignard, F.; Bertrand, O.; Caclin, A.; Mattout, J. Empirical Bayes evaluation of fused EEG-MEG source reconstruction: Application to auditory mismatch evoked responses. *NeuroImage* **2021**, *226*, 117468. [CrossRef]
23. Kuznetsova, A.; Nurislamova, Y.; Ossadtchi, A. Modified covariance beamformer for solving MEG inverse problem in the environment with correlated sources. *NeuroImage* **2021**, *228*, 117677. [CrossRef]
24. Samuelsson, J.G.; Peled, N.; Mamashli, F.; Ahveninen, J.; Hämäläinen, M.S. Spatial fidelity of MEG/EEG source estimates: A general evaluation approach. *Neuroimage* **2021**, *224*, 117430. [CrossRef]
25. Cai, C.; Sekihara, K.; Nagarajan, S.S. Hierarchical multiscale Bayesian algorithm for robust MEG/EEG source reconstruction. *NeuroImage* **2018**, *183*, 698–715. [CrossRef]
26. Hindriks, R. A methodological framework for inverse-modeling of propagating cortical activity using MEG/EEG. *NeuroImage* **2020**, *223*, 117345. [CrossRef]
27. Janati, H.; Bazeille, T.; Thirion, B.; Cuturi, M.; Gramfort, A. Multi-subject MEG/EEG source imaging with sparse multi-task regression. *NeuroImage* **2020**, *220*, 116847. [CrossRef]
28. Grova, C.; Aiguabella, M.; Zelmann, R.; Lina, J.M.; Hall, J.A.; Kobayashi, E. Intracranial EEG potentials estimated from MEG sources: A new approach to correlate MEG and iEEG data in epilepsy. *Hum. Brain Mapp.* **2016**, *37*, 1661–1683. [CrossRef] [PubMed]
29. Pellegrino, G.; Hedrich, T.; Chowdhury, R.; Hall, J.A.; Lina, J.M.; Dubeau, F.; Kobayashi, E.; Grova, C. Source localization of the seizure onset zone from ictal EEG/MEG data. *Hum. Brain Mapp.* **2016**, *37*, 2528–2546. [CrossRef]
30. Pellegrino, G.; Hedrich, T.; Chowdhury, R.A.; Hall, J.A.; Dubeau, F.; Lina, J.M.; Kobayashi, E.; Grova, C. Clinical yield of magnetoencephalography distributed source imaging in epilepsy: A comparison with equivalent current dipole method. *Hum. Brain Mapp.* **2018**, *39*, 218–231. [CrossRef] [PubMed]
31. Pellegrino, G.; Hedrich, T.; Porras-Bettancourt, M.; Lina, J.M.; Aydin, Ü.; Hall, J.; Grova, C.; Kobayashi, E. Accuracy and spatial properties of distributed magnetic source imaging techniques in the investigation of focal epilepsy patients. *Hum. Brain Mapp.* **2020**, *41*, 3019–3033. [CrossRef] [PubMed]
32. Chowdhury, R.A.; Pellegrino, G.; Aydin, Ü.; Lina, J.M.; Dubeau, F.; Kobayashi, E.; Grova, C. Reproducibility of EEG-MEG fusion source analysis of interictal spikes: Relevance in presurgical evaluation of epilepsy. *Hum. Brain Mapp.* **2018**, *39*, 880–901. [CrossRef]
33. Sarvas, J. Basic mathematical and electromagnetic concepts of the biomagnetic inverse problem. *Phys. Med. Biol.* **1987**, *32*, 11. [CrossRef] [PubMed]
34. Reimann, M.; Doerner, K.; Hartl, R.F. D-ants: Savings based ants divide and conquer the vehicle routing problem. *Comput. Oper. Res.* **2004**, *31*, 563–591. [CrossRef]
35. Shmygelska, A.; Hoos, H.H. An ant colony optimisation algorithm for the 2D and 3D hydrophobic polar protein folding problem. *BMC Bioinform.* **2005**, *6*, 30. [CrossRef]
36. Singer, S.; Nelder, J. Nelder-mead algorithm. *Scholarpedia* **2009**, *4*, 2928. [CrossRef]
37. Gutjahr, W.J. A graph-based ant system and its convergence. *Future Gener. Comput. Syst.* **2000**, *16*, 873–888. [CrossRef]
38. Gutjahr, W.J. ACO algorithms with guaranteed convergence to the optimal solution. *Inf. Process. Lett.* **2002**, *82*, 145–153. [CrossRef]
39. Henson, R.; Mattout, J.; Phillips, C.; Friston, K.J. Selecting forward models for MEG source-reconstruction using model-evidence. *Neuroimage* **2009**, *46*, 168–176. [CrossRef]
40. Henson, R.N.; Mattout, J.; Singh, K.D.; Barnes, G.R.; Hillebrand, A.; Friston, K. Population-level inferences for distributed MEG source localization under multiple constraints: Application to face-evoked fields. *Neuroimage* **2007**, *38*, 422–438. [CrossRef] [PubMed]
41. Brett, M.; Penny, W.; Kiebel, S. Introduction to random field theory. *Hum. Brain Funct.* **2003**, *2*. [CrossRef]
42. Vehtari, A.; Gelman, A.; Sivula, T.; Jylänki, P.; Tran, D.; Sahai, S.; Blomstedt, P.; Cunningham, J.P.; Schiminovich, D.; Robert, C.P. Expectation Propagation as a Way of Life: A Framework for Bayesian Inference on Partitioned Data. *J. Mach. Learn. Res.* **2020**, *21*, 1–53.