Supplement to "Communication-Avoiding Optimization Methods for Distributed Massive-Scale Sparse Inverse Covariance Estimation"

Penporn Koanantakool[1,2,*] **Alnur Ali**[3] **Ariful Azad**[2] **Aydın Buluç**[2,1]
Dmitriy Morozov[2,5] **Leonid Oliker**[2] **Katherine Yelick**[1,2] **Sang-Yun Oh**[4,2]
[1]Department of Electrical Engineering and Computer Sciences, UC Berkeley
[2]Computational Research Division, Lawrence Berkeley National Laboratory
[3]Machine Learning Department, Carnegie Mellon University
[4]Department of Statistics and Applied Probability, UC Santa Barbara
[5]Berkeley Institute for Data Science, UC Berkeley

This document contains proofs and supplementary details for the paper "Communication-Avoiding Optimization Methods for Distributed Massive-Scale Sparse Inverse Covariance Estimation". All section, equation, table, and figure numbers in this supplementary document are preceded by the letter "S" (all section, equation, table, and figure numbers without an "S" refer to the main paper).

S.1 PSEUDOCODE FOR Obs

The pseudocode for the Obs variant of HP-CONCORD, omitted from the main paper due to space constraints, is presented below in Algorithm S.1. The main differences from Algorithm 1 in the main paper are highlighted in blue.

Algorithm S.1 The Obs variant of HP-CONCORD, for computing a sparse inverse covariance matrix estimate.

```
Input: data matrix $X \in \mathbf{R}^{n\times p}$; tuning parameters $\lambda_1, \lambda_2 \geq 0$; optimization tolerance $\epsilon > 0$
Output: estimate $\hat{\Omega} \in \mathbf{R}^{p\times p}$ of the underlying inverse covariance matrix $\Omega^0$
 1: $\Omega^{(0)} \leftarrow I$
 2: Compute $Y^{(0)} \leftarrow \Omega^{(0)} X^T$          ▷ Compute via a distributed sparse-dense matrix multiplication
 3: for $k = 0, 1, 2, \ldots$
 4:     Compute $Z^{(k)} \leftarrow Y^{(k)} X$          ▷ Compute via a distributed dense-dense matrix multiplication
 5:     Form $(Z^{(k)})^T$          ▷ Form via a distributed matrix transpose
 6:     $G^{(k)} \leftarrow -(\Omega_D^{(k)})^{-1} + \frac{1}{2}((Z^{(k)})^T + Z^{(k)}) + \lambda_2 \Omega^{(k)}$          ▷ Use $Z^{(k)}, (Z^{(k)})^T$
 7:     $g(\Omega^{(k)}) \leftarrow -2\sum_i \log(\Omega_{ii}^{(k)}) + \frac{1}{n}\|Y^{(k)}\|_F^2 + \frac{\lambda_2}{2}\|\Omega^{(k)}\|_F^2$          ▷ Use $Y^{(k)}$; see text for details
 8:     for $\tau = 1, \frac{1}{2}, \frac{1}{4}, \ldots$
 9:         $\Omega^{(k+1)} \leftarrow \mathcal{S}_{\tau\lambda_1}(\Omega^{(k)} - \tau G^{(k)})$          ▷ Apply the soft-thresholding operator, $\mathcal{S}_{\tau\lambda_1}$
10:         Compute $Y^{(k+1)} \leftarrow \Omega^{(k+1)} X^T$          ▷ Compute via a distributed sparse-dense matrix multiplication
11:         $g(\Omega^{(k+1)}) \leftarrow -2\sum_i \log(\Omega_{ii}^{(k+1)}) + \frac{1}{n}\|Y^{(k+1)}\|_F^2 + \frac{\lambda_2}{2}\|\Omega^{(k+1)}\|_F^2$          ▷ Use $Y^{(k+1)}$; see text for details
12:     until $g(\Omega^{(k+1)}) \leq g(\Omega^{(k)}) - \mathbf{tr}((\Omega^{(k)} - \Omega^{(k+1)})^T G^{(k)}) + \frac{1}{2\tau}\|\Omega^{(k)} - \Omega^{(k+1)}\|_F^2$          ▷ See text for details
13: until a stopping criterion is satisfied, using $\epsilon$
14: return the estimate $\hat{\Omega} \leftarrow \Omega^{(k)}$
```

[*] Now at Google Brain.

S.2 COMPUTATIONAL COST COMPARISON FOR Cov VS. Obs

Cov is preferred over Obs when the flop count for Cov is less than that for Obs (from Lemma 3.1 in the main paper), *i.e.*,

$$\begin{aligned}2np^2 + 2dp^2(st+1) &< 2np^2s + 2dnp(st+1)\\ 2dp(st+1)(p-n) &< 2np^2(s-1)\\ d(st+1)(p-n) &< np(s-1).\end{aligned} \tag{S.1}$$

We relax the comparison a little by plugging in $st < st+1$ and $s > s-1$:

$$\begin{aligned}dst(p-n) &< nps\\ \frac{d}{p} &< \frac{n}{p-n}\cdot\frac{1}{t}.\end{aligned} \tag{S.2}$$

Let $r_{\text{obs}} = n/p$ be the ratio of the number of observations to the number of features and let $r_{\text{nnz}} = dp/p^2 = d/p$ be the *average* percent nonzeroes of Ω throughout all iterations, $0 < r_{\text{obs}}, r_{\text{nnz}} \leq 1$, we can reformulate (S.2) as

$$r_{\text{nnz}} < \frac{r_{\text{obs}}}{1-r_{\text{obs}}}\cdot\frac{1}{t}.$$

Above, $r_{\text{obs}}/(1-r_{\text{obs}})$ is an increasing function. The closer n is to p, the higher r_{nnz} can be for Cov to still require less flops than Obs. For example, assume $t = 10$ (we observed 5-15 line search iterations per one proximal gradient iteration in practice). Substituting $r_{\text{obs}} = 0.01$, 0.1, and 0.25 give $r_{\text{nnz}} < 0.001, 0.011$, and 0.033, respectively. Applications with *average* percent nonzeroes of Ω (throughout all iterations) less than $0.1\%, 1.1\%$, and 3.3% should benefit from Cov in these cases.

S.3 ADDITIONAL DETAILS ON OUR 1.5D MATRIX MULTIPLICATION ALGORITHM

This subsection details how Algorithm 3 in the main paper computes $C = AB$. Our algorithm partitions all matrices in a 1D layout. We pick one operand to move around (A or B) and fixing the other operand and C stationary. We will call the rotating matrix R and the other fixed operand F. The notation we use is summarized in the table below. Our indexing is one-based and cyclic.

Notation	**Meaning**
P	Total number of processors
R	Matrix operand being rotated (A or B)
F	Matrix operand being fixed (B or A)
c_R	Replication factor of R
c_F	Replication factor of F and C
G_R	Logical grid of processors of size P/c_R teams by c_R layers
G_F	Logical grid of processors of size P/c_F teams by c_F layers

S.3.1 Data layout

We partition R in 1D into P/c_R equal parts: $R(1), \ldots, R(P/c_R)$ and let processors $G_R(i,:)$ own $R(i)$. There are c_R processors in each team $G_R(i,:)$ so each $R(i)$ is stored c_R times across all processors, hence why we call c_R the replication factor of R. Likewise, we partition F and C in 1D into P/c_F equal parts: $F(1), \ldots, F(P/c_F)$ and $C(1), \ldots, C(P/c_F)$. We let processors $G_F(j,:)$ own $F(j)$ and $C(j)$.

S.3.2 Multiplication

Team $G_F(j,:)$ is responsible for computing $C(j)$ together. Each team member computes appropriate multiplication between the fixed matrix $F(j)$ with different parts of R. For convenience, we reproduce the pseudocode for Algorithm 3 from the main paper below.

Algorithm S.2 Our 1.5D matrix multiplication algorithm.

```
1: for each $G_R(i,\ell_R) = G_F(j,\ell_F)$, in parallel, do
2:     $\delta \leftarrow \min(\ell_F,\ell_R) \cdot \max(1, c_F/c_R)$
3:     Shift $R$ by $\delta$.
4:     for $\frac{p}{c_F c_R}$ rounds do
5:         Calculate local $C = AB$
6:         Shift $R$ by $c_F$
7:     end for
8:     SumReduce/Allgather $C$ between $G_F(j,:)$
9: end for
```

Line 2 calculates an offset δ that each member would shift R initially to all get different parts of R. Line 3 performs the initial shift. Then we alternate between multiplying the local matrices and shifting R by c_F to get a new part of R. There are P/c_R different parts of R and each team calculates c_F parts at once (one part per each team member). The algorithm needs to run for $P/(c_R c_F)$ rounds.

Figure S.1 shows an example when we partition A in 1D block row layout, B and C in 1D block column layout, and rotate A, on 16 processors. Figure S.1a shows the situation when $c_A \leq c_B$. The first row shows the original layout of A and B. Written on each matrix part are the processor ranks. $c_A = 4$ and $c_B = 2$. The second row shows Round 0, where each processor performs an initial shift so that $G_B(j,:)$ as a whole would multiply $A(jc_B/c_A)$ to $A(jc_B/c_A + c_B)$ with $B(j)$. The next round the team moves to the next c_B blocks of A and finish the rest of the multiplications. Figure S.1b shows what happens when $c_A > c_B$. Figure S.2 illustrates how the algorithm works when we shift B instead of A.

S.3.3 Communication costs

Let $\mathbf{nnz}(\cdot)$ denote the number of nonzereos of a matrix, sparse or dense. There are $p/(c_R c_F)$ rounds, each round each processor sends a message. The message size is $\mathbf{nnz}(R)/(P/c_R) = c_R\,\mathbf{nnz}(R)/P$. The total number of messages and words are

$$L_{1.5\text{D}} = \frac{P}{c_R c_F} \text{ messages,}$$
$$W_{1.5\text{D}} = \frac{P}{c_R c_F} \cdot \frac{c_R\,\mathbf{nnz}(R)}{P} = \frac{\mathbf{nnz}(R)}{c_F} \text{ words.}$$

S.3.4 Transposing the resulting matrix C

Normal 1D partitioning without replication requires all P processors to talk to all other processors in a transpose. Replication helps us limit the number of processors each processor has to exchange matrices with. For example, in Figure S.1a, processor 0 has to exchange sub-matrices with processors 2, 8, and 10 to do a transpose, not all processors anymore. For each team $G_F(j,:)$, there are P/c_R block rows of $C(j)$ to calculate, so each processor $G_F(j,\ell_F)$ calculates $P/(c_R c_F)$ submatrices of $C(j)$. When being transposed, each submatrix will span $\max(c_R/c_F, c_F/c_R)$ submatrices from other teams. The number of processors each processor has to communicate to in a transpose is

$$\frac{P}{c_R c_F} \max\left(\frac{c_R}{c_F}, \frac{c_F}{c_R}\right) = \max\left(\frac{P}{c_F^2}, \frac{P}{c_R^2}\right).$$

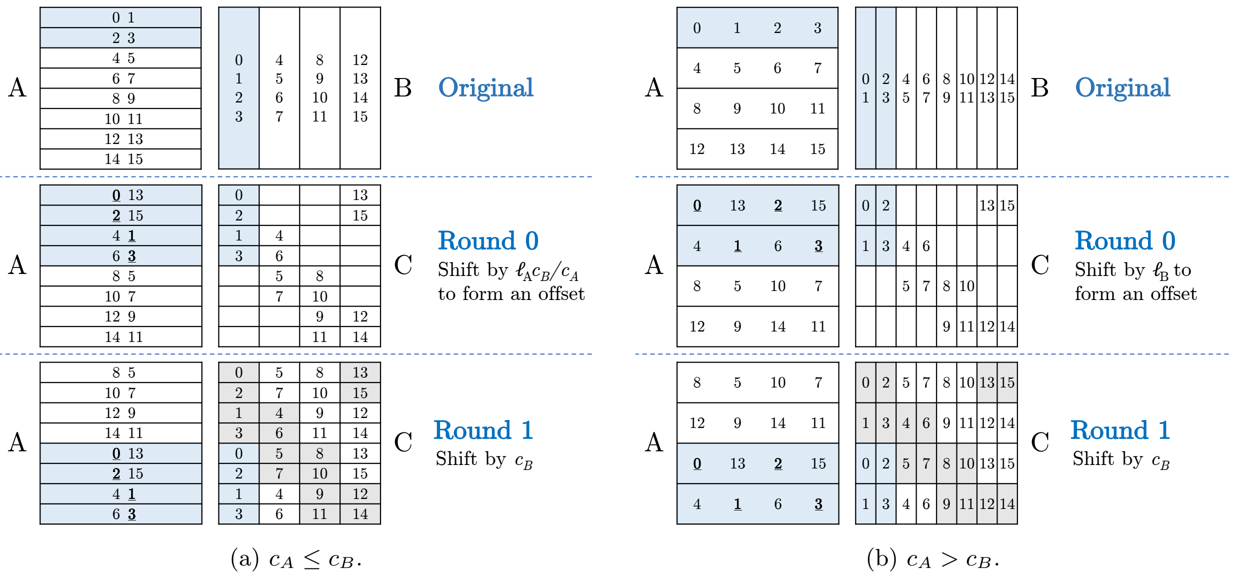

A
B
Original
C
Round 0
Shift by $\ell_A c_B/c_A$ to form an offset
Round 1
Shift by c_B
(a) $c_A \leq c_B$.
Shift by ℓ_B to form an offset
(b) $c_A > c_B$.

Figure S.1: Processors $G_B(j,:)$ compute $C(j)$ together. The numbers on the matrix parts are the ranks of the processors that it resides on. Here, $p = 16$ and (c_A, c_B) is $(2,4)$ in (a) and $(4,2)$ in (b). The first line shows the original layouts of A and B. The second line (Round 0) shifts A by δ to compute the first c_B blocks of $C(j)$. The third line (Round 1) shifts A by c_B and computes the rest of C.

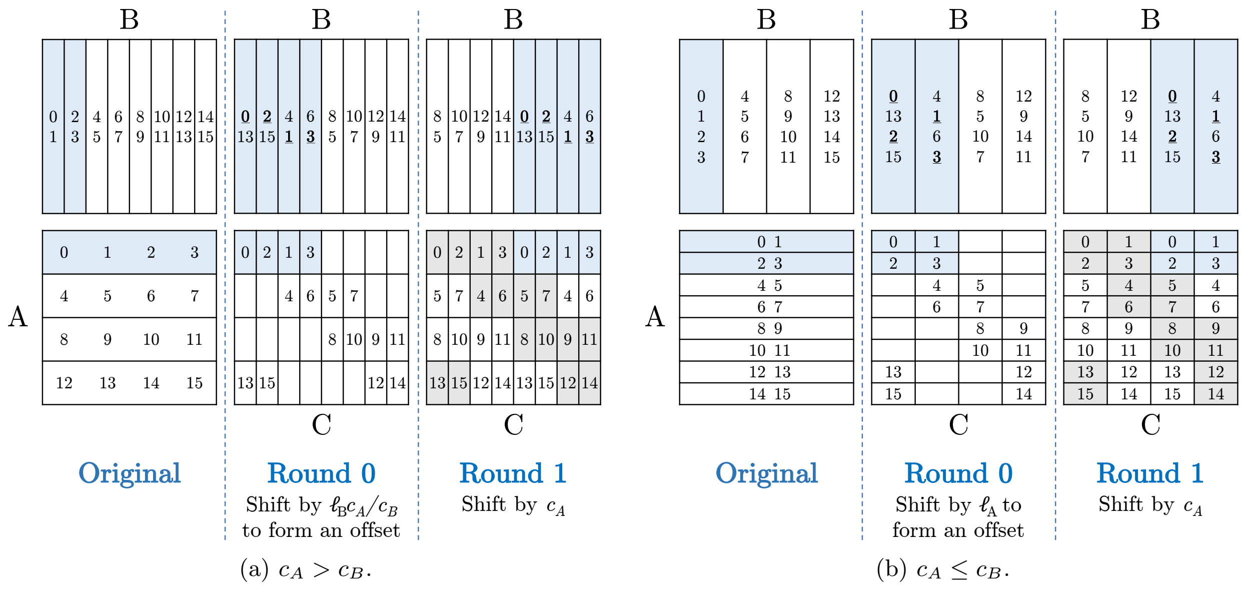

B
A
C
Original
Round 0
Shift by $\ell_B c_A/c_B$ to form an offset
Round 1
Shift by c_A
(a) $c_A > c_B$.
Shift by ℓ_A to form an offset
(b) $c_A \leq c_B$.

Figure S.2: Processors $G_A(i,:)$ compute $C(i)$ together. The numbers on the matrix parts are the ranks of the processors that it resides on. Here, $p = 16$ and (c_A, c_B) is $(4,2)$ in (a) and $(2,4)$ in (b). The Original column shows the original layouts of A and B. Round 0 shifts B by δ to compute the first c_A blocks of $C(i,:)$. Round 1 shifts B by c_A and computes the rest of C.

Each processor holds $\mathbf{nnz}(C)/(P/c_F)$ words of C, split into P/c_R blocks. Therefore, the number of words each block to be transposed has is

$$\mathbf{nnz}(C) \cdot \frac{c_R c_F}{P^2}.$$

An all-to-all communication between P processors where everyone sends and receives w words from everyone takes $O(\log P)$ messages and $O(wP \log P)$ words. Therefore, the costs of transposing the resulting matrix C from Algorithm 3 are

$$\begin{aligned} L_{\text{xpose}} &= O\left(\log \max\left(\frac{P}{c_F^2}, \frac{P}{c_R^2}\right)\right) \text{ messages,} \\ W_{\text{xpose}} &= O\left(\frac{\mathbf{nnz}(C) c_R c_F}{P^2} \max\left(\frac{P}{c_F^2}, \frac{P}{c_R^2}\right) \log \max\left(\frac{P}{c_F^2}, \frac{P}{c_R^2}\right)\right) \text{ words.} \end{aligned}$$

S.4 ADDITIONAL DETAILS, FIGURES, AND TABLES FOR THE fMRI EXPERIMENTS

S.4.1 Further details on the fMRI data

Here, we present further details on the fMRI data, and in particular the sample covariance matrix, we use. The sample covariance matrix we use was generated in the following way (c.f. Figure 2 of [11]). First, 1,200 subjects were put into a state-of-the-art fMRI machine and measurements were taken without stimulating the subjects, every 0.7 seconds for an hour, at 2 millimeter × 2 millimeter × 2 millimeter cubes/voxels spread evenly throughout the cerebral cortex. Next, as fMRI data is typically very noisy, a significant amount of post-processing was done to denoise the data, ultimately leading to a data matrix X with dimensions $n \approx 6{,}171{,}400$, $p = 91{,}282$. To further reduce the level of noise, the columns of the data matrix were then averaged over the 1,200 subjects, leading to a data matrix with dimensions $n \approx 5{,}142$, $p = 91{,}282$, from which the sample covariance matrix was finally computed.

S.4.2 Related work

We mention some work that is related to our approach, noting that most other works (i) do not use the latest available fMRI data, (ii) are not particularly scalable, and/or (iii) use nonconvex estimators, which can lead to interpretability issues. The work of [10] applies the “SPACE” estimator of [7], except augmented with an additional squared ℓ_2-norm penalty, to low-dimensional fMRI data; while certainly related to our approach, this method unfortunately suffers from all three of the aforementioned issues. The work of [6] applies the graphical lasso algorithm of [5] to older fMRI data, except with an ℓ_0-“norm” penalty instead of the usual ℓ_1-norm penalty, making the corresponding optimization problem nonconvex. The work of [2] applies a pseudolikelihood-based estimator with a deep neural network as the predictive primitive (instead of, say, a lasso regression) to fMRI data obtained from subjects with autism instead of the broader population. Lastly, the works of [8, 9] focus on scalable methods for structure recovery from fMRI data, rather than for the broader problem of sparse inverse covariance estimation.

S.4.3 Further details on the sparsity patterns of the HP-CONCORD estimates

It is interesting to analyze the sparsity patterns of the HP-CONCORD and sample covariance matrix estimates yielding the best clusterings; these are presented in the left-most column of Table 2. Three features here are striking: (i) the HP-CONCORD estimates possess a block diagonal structure, where the blocks turn out to correspond to the left and right hemispheres; (ii) furthermore, the sparsity patterns of the blocks themselves turn out to correspond to the (spatially) closest voxels (the supplement provides details); (iii) although the sparsity patterns of the HP-CONCORD and sample covariance matrix estimates appear vaguely similar, the subtle differences between them drive the (significant) differences in the clusterings. We emphasize that these features arise naturally, without being hard-coded into our method.

The sparsity patterns of the diagonal blocks of the HP-CONCORD estimates turn out to correspond to the (spatially) closest voxels on the left and right hemispheres; here we provide some further details. In Figure S.3, we present the sparsity pattern of the HP-CONCORD estimate attaining the best clustering, for the left hemisphere, in the left column of Table 2 in the main paper; in Figure S.3, we also present the sparsity pattern of a matrix we constructed, where the (i, j)th entry of the matrix is the great-circle distance between the voxels i and j, after retaining only 0.1% of closest voxels. The sparsity patterns of the distance matrix and HP-CONCORD estimates indeed look visually similar, suggesting that the best HP-CONCORD estimate has recovered some of the spatial signal in the data, without being "told" to do so. Inspecting the right hemisphere conveys the same message.

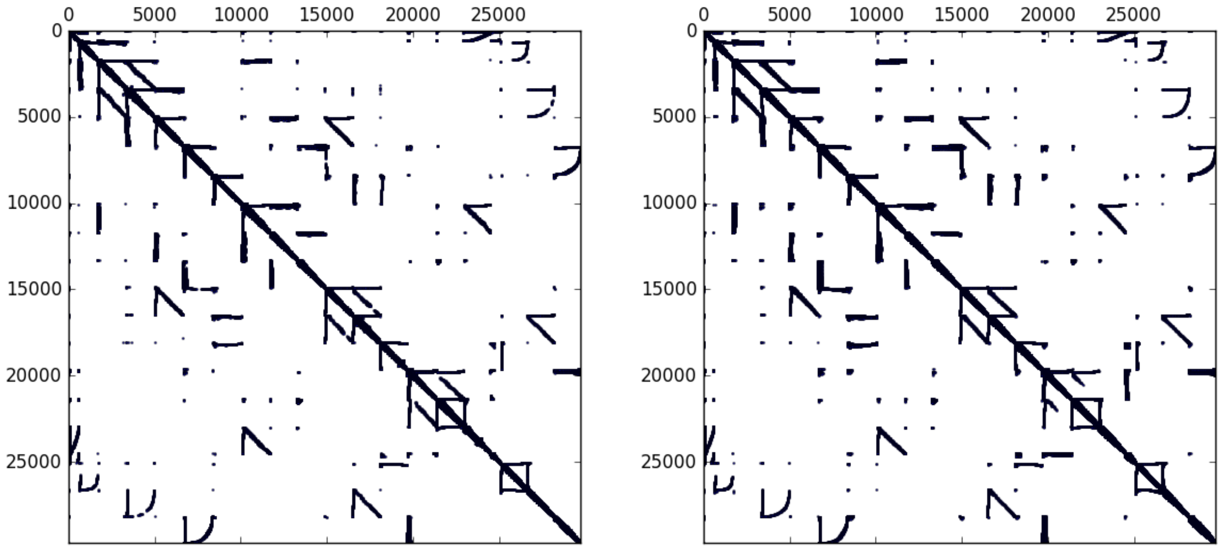


Figure S.3: Left: the sparsity pattern of the HP-CONCORD estimate attaining the best clustering in the left column of Table 2 in the main paper, where we have only plotted the 29,696 coordinates belonging to the left hemisphere. Right: the sparsity pattern of a $(91,282 \times 91,282)$-dimensional (symmetric) matrix we constructed, whose (i, j)th entry is the great-circle distance between the voxels i and j, where we have (again) only plotted the 29,696 coordinates belonging to the left hemisphere, and additionally retained just the 0.1% of closest voxels. In both plots, black indicates a nonzero entry.

S.4.4 Details on persistent homology-based clustering algorithm

We map the degree of a vertex in the inverse covariance graph, described by matrix $\Omega^{p \times p}$, onto the surface of a brain. We thus obtain a function $f : S \to \mathbb{Z}$, where S is the triangulation of the cortical surface. We apply the watershed algorithm [3] to f by sweeping the vertices from the highest value to the lowest. We start a new label if the vertex has no labeled neighbors in S. If it does, we propagate the label with the maximum starting value.

The resulting parcellation is usually too fine: every local maximum of f produces a new label. We use the theory of persistent homology [4] to coarsen the parcellation. During the sweep, we build the dual graph G of the labels. When we start a new label l at a vertex u, we add l to the graph and assign to it the value $f(u)$. When two labels, l_1 and l_2, that fall in different components of G, meet at a vertex v during the sweep, we add an edge (l_1, l_2) to G. We find the maximum values, a_1 and a_2, assigned to any vertex in the components of l_1 and l_2 in G, and assign to the new edge the value $\min\{a_1, a_2\} - f(v)$. (It's not difficult to verify that this is exactly the persistence of the vertex v.)

Once we construct the dual graph G, given a simplification threshold ε, we treat the connected components of the subgraph of G induced by the edges with values at most ε as the new parcels. As we increase ε, the parcels merge, and the parcellation gets coarser.

S.4.5 The (modified) Jaccard score

To quantitatively compare clusterings, we consider a variation of the standard Jaccard score,

$$\mathrm{Sim}(\mathcal{C}_1, \mathcal{C}_2) = \frac{1}{\max(k, \ell)} \cdot \sum_{(i,j)\in\mathcal{E}} W_{ij}, \tag{S.3}$$

where $\mathcal{C}_1 = \{A_1, \dots, A_k\}$ and $\mathcal{C}_2 = \{B_1, \dots, B_\ell\}$ are two clusterings; above, $\mathcal{E} \subseteq \{1, \dots, k\} \times \{1, \dots, \ell\}$ is a maximum weighted edge covering in a (weighted) bipartite graph, where the vertices on a side of the graph correspond to the clusters in a clustering, and the edge weights $W_{ij},\ i = 1, \dots, m,\ j = 1, \dots, \ell$, are the usual Jaccard scores given by $\frac{|A_i \cap B_j|}{|A_i \cup B_j|}$. The use of the edge covering here resolves various complications that arise when comparing clusterings of different sizes; the $\frac{1}{\max(k,\ell)}$ term in (S.3) can be thought of as a normalizing constant Finally, to compute the edge covering, we use the algorithm of [1].

S.4.6 Expanded set of results

Here, we provide an expanded set of figures and tables from the experiments with (i) various penalty values of HP-CONCORD (*i.e.*, λ_1 and λ_2), (ii) two parts of the brain (left and right hemisphere), (iii) two clustering methods, taking in the partial correlation graph from HP-CONCORD as inputs (persistent homology and Louvain methods), and (iv) two clustering coarseness levels (more clusters and fewer clusters). For convenience, we outline where to find each of our results in the table below.

Method		**Left hemisphere**		**Right hemisphere**	
		Clusterings	Jaccard scores	Clusterings	Jaccard scores
Persistent homology	with $\varepsilon = 3$ (fewer clusters)	See Table S.1	See Table S.9	See Table S.2	See Table S.10
	with $\varepsilon = 0$ (more clusters)	See Table S.3	See Table S.11	See Table S.4	See Table S.12
Louvain	with $k = 0$ (fewer clusters)	See Table S.5	See Table S.13	See Table S.6	See Table S.14
	with $k =$ max. # clusters from Louvain	See Table S.7	See Table S.15	See Table S.8	See Table S.16

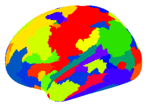
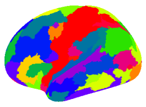
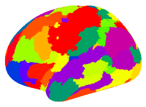
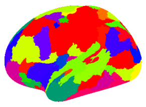
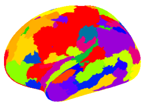
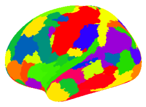
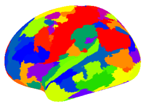
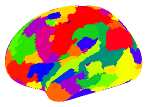
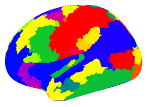
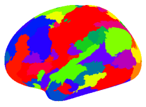
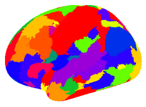
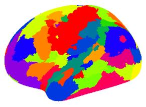
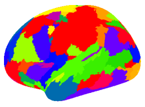
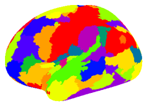
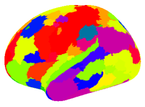
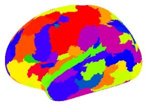
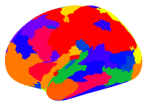
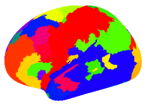
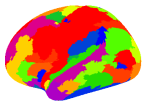
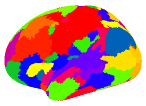
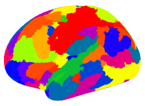
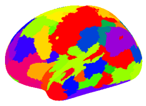
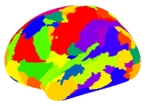
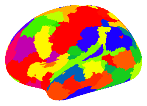
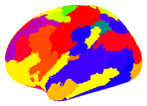
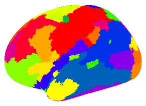
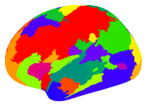
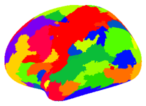
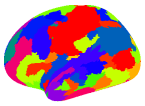
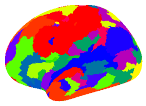
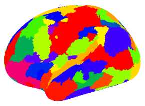
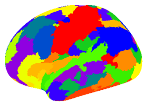
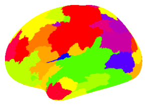
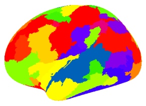
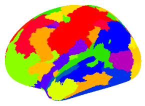
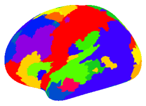
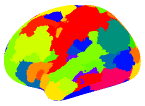
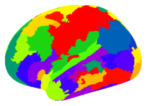
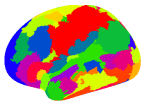
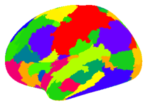
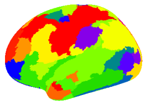
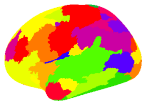
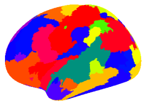
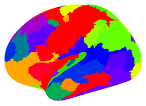
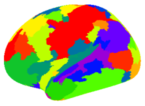
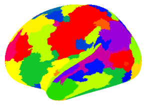
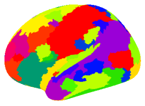
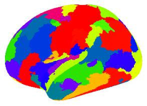
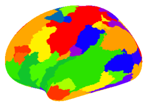
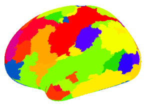
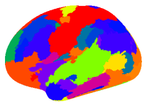
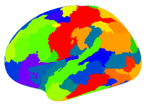
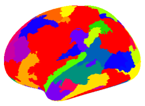
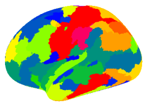
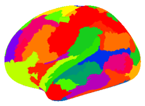
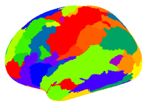
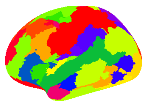
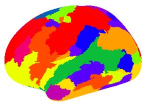
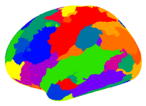
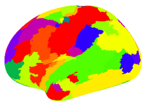
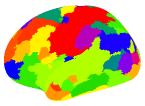
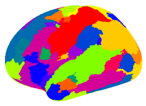
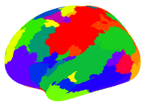
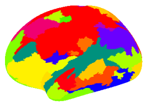
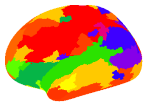
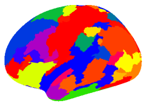
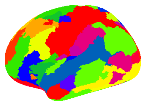
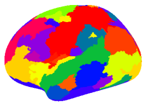
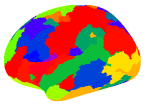
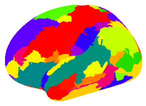
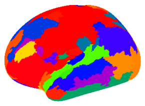
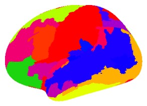
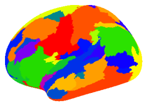
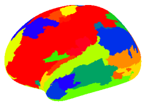
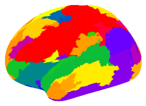
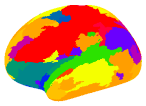
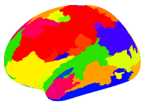
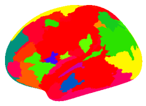
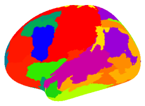
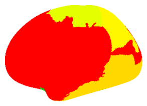
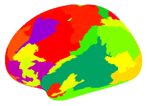
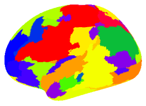
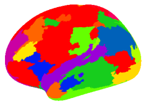
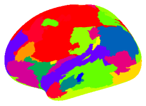
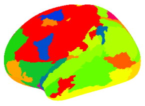
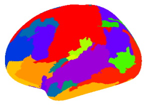
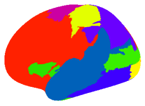
λ_1 \ λ_2	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48								
0.5								
0.5208								
0.5425								
0.5651								
0.5887								
0.6132								
0.6388								
0.6654								
0.6931								
0.722								—

Table S.1: The clusterings, for the *left* hemisphere, generated by HP-CONCORD followed by the *persistent homology* method, at the tuning parameter values: $\varepsilon = 3$ (generally corresponding to *fewer* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. Table S.9 presents the Jaccard scores (S.3) for these clusterings. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

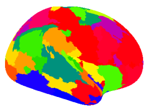
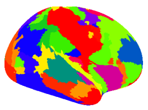
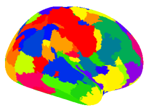
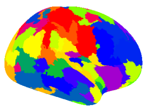
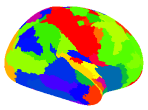
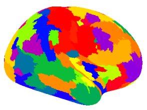
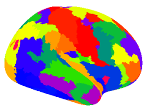
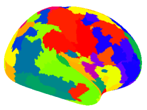
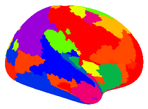
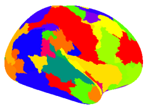
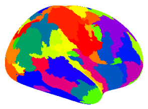
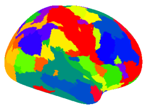
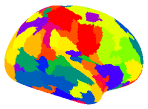
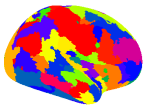
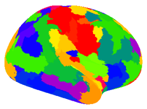
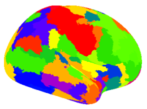
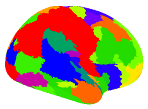
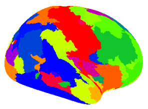
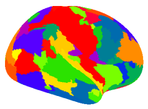
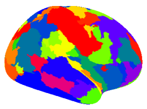
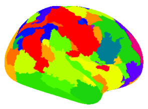
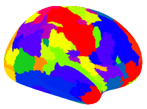
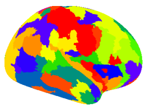
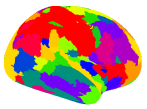
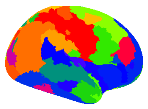
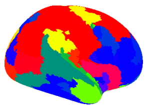
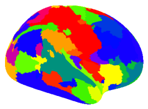
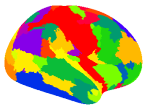
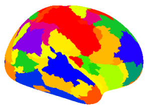
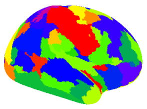
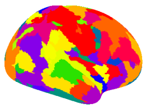
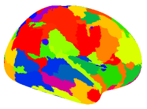
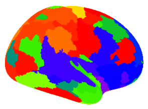
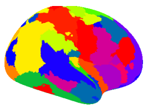
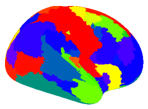
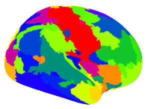
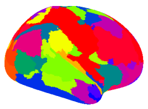
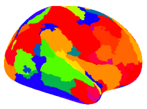
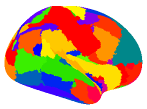
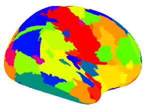
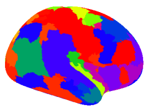
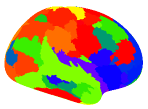
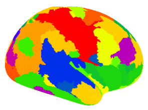
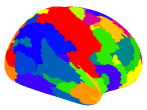
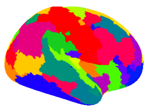
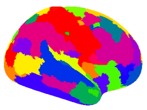
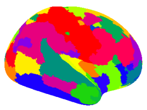
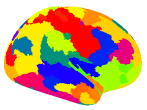
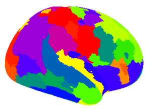
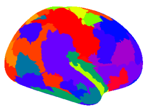
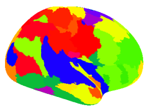
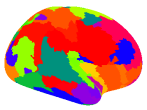
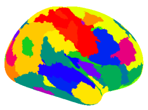
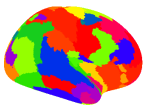
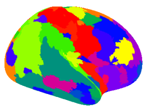
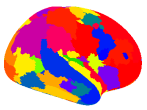
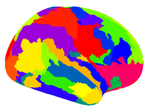
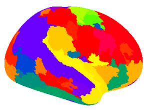
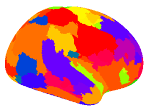
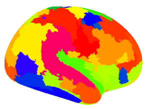
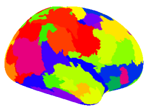
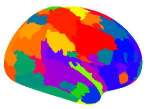
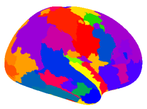
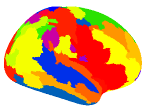
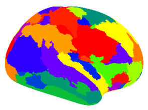
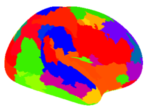
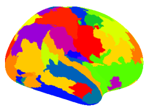
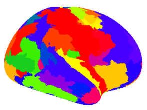
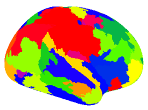
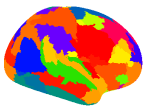
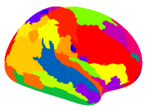
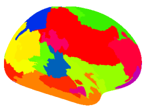
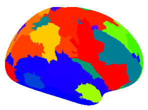
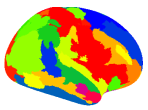
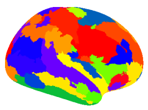
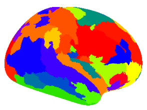
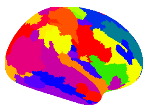
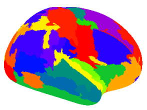
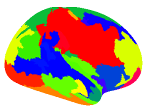
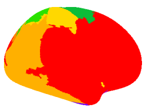
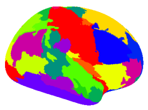
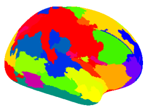
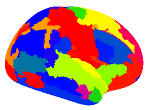
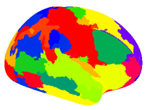
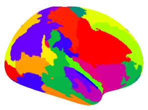
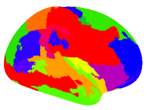
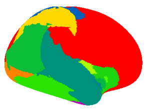
λ_1 \ λ_2	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48								
0.5								
0.5208								
0.5425								
0.5651								
0.5887								
0.6132								
0.6388								
0.6654								
0.6931								
0.722								—

Table S.2: The clusterings, for the *right* hemisphere, generated by HP-CONCORD followed by the *persistent homology* method, at the tuning parameter values: $\varepsilon = 3$ (generally corresponding to *fewer* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. Table S.10 presents the Jaccard scores (S.3) for these clusterings. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

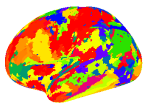
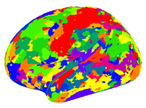
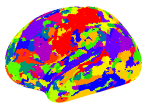
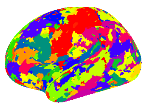
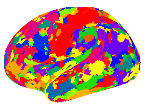
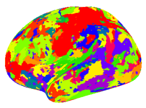
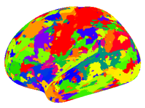
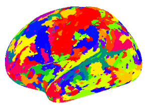
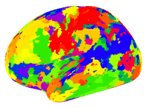
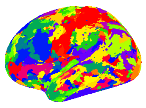
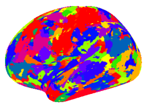
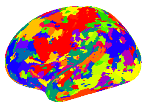
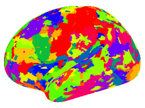
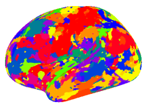
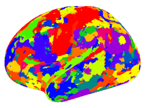
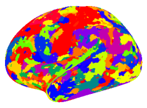
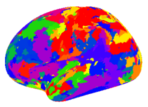
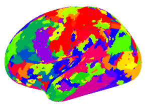
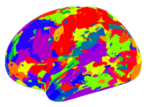
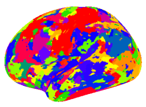
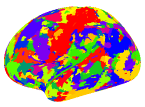
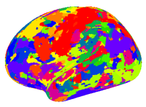
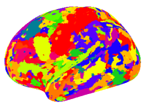
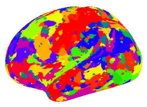
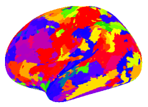
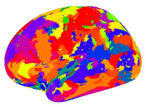
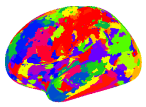
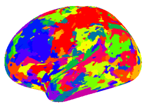
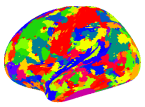
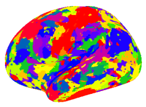
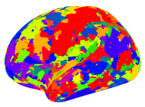
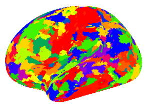
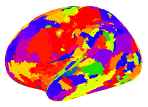
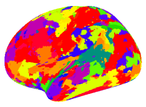
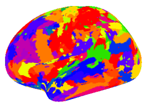
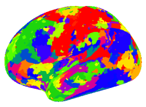
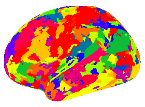
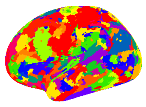
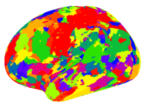
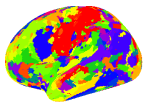
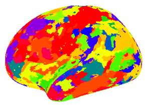
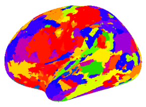
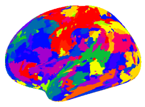
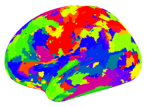
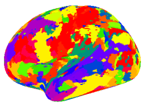
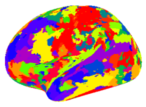
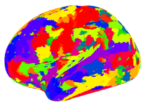
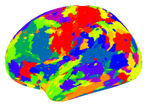
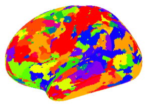
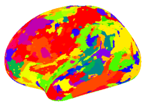
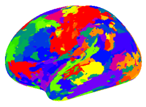
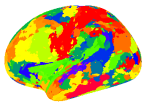
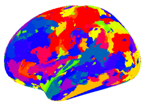
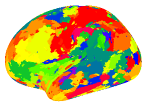
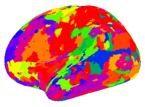
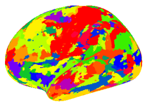
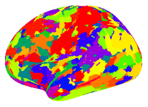
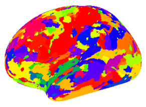
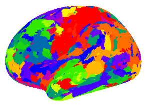
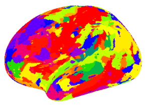
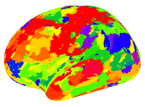
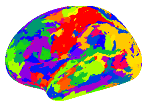
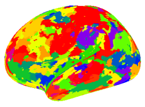
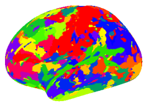
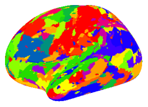
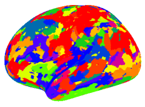
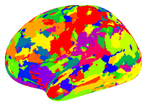
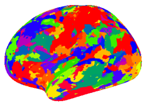
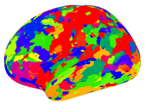
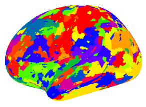
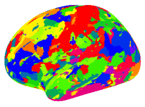
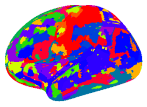
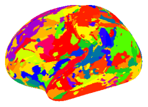
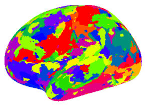
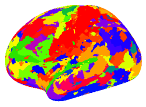
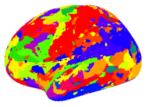
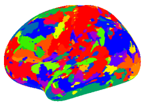
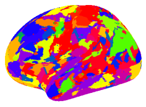
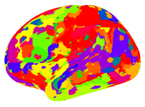
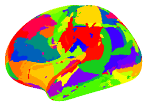
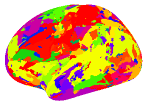
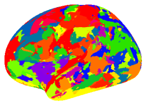
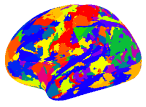
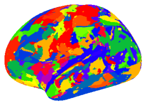
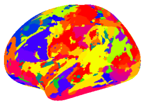
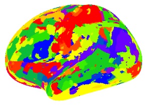
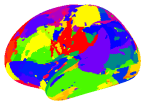
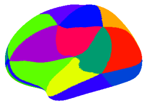
λ_1 \ λ_2	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48								
0.5								
0.5208								
0.5425								
0.5651								
0.5887								
0.6132								
0.6388								
0.6654								
0.6931								
0.722								

Table S.3: The clusterings, for the *left* hemisphere, generated by HP-CONCORD followed by the *persistent homology* method, at the tuning parameter values: $\varepsilon = 3$ (generally corresponding to *more* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. Table S.11 presents the Jaccard scores (S.3) for these clusterings. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

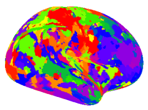
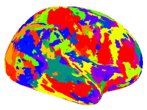
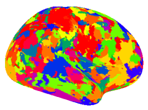
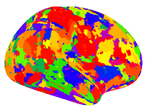
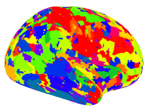
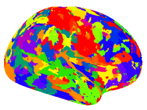
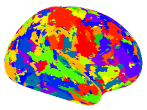
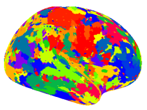
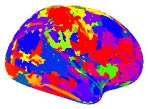
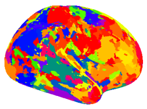
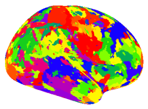
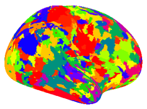
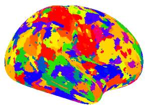
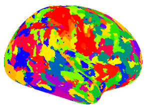
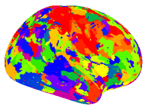
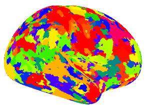
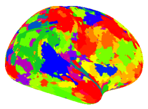
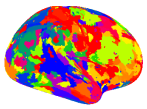
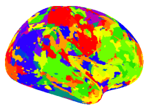
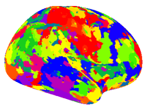
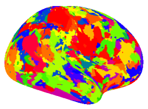
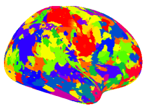
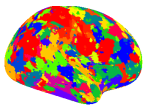
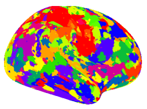
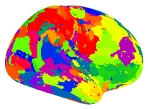
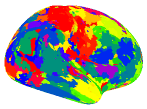
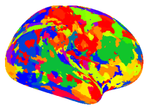
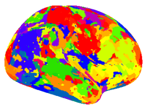
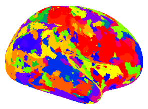
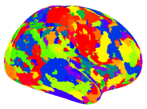
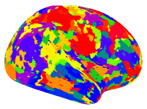
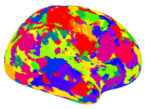
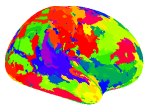
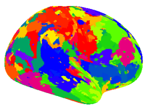
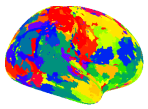
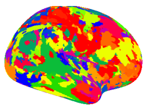
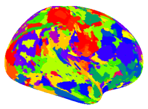
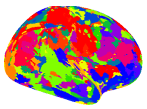
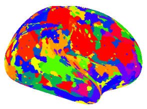
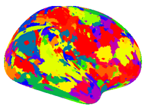
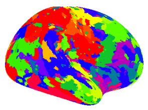
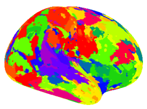
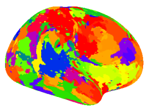
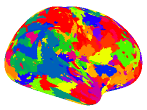
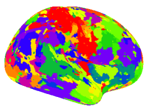
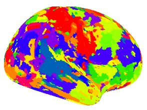
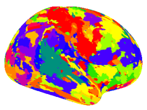
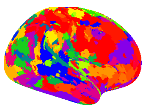
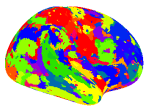
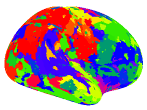
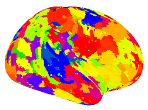
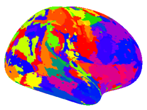
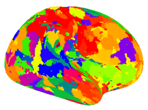
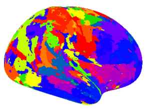
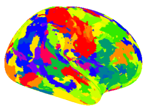
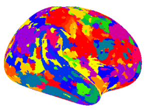
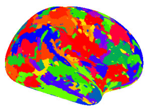
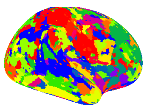
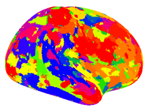
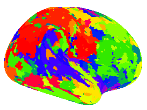
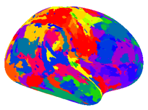
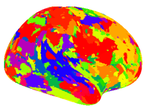
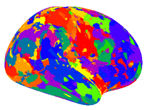
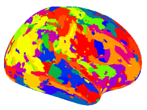
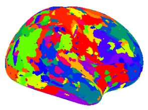
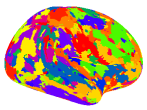
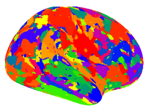
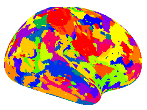
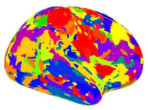
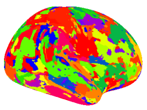
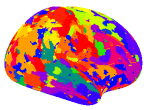
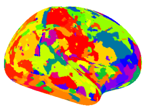
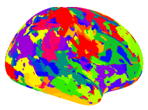
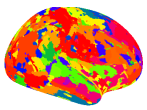
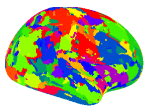
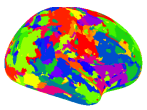
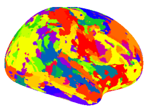
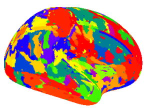
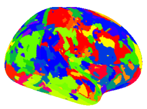
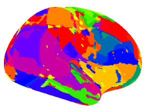
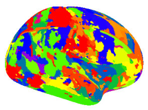
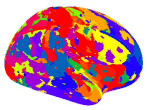
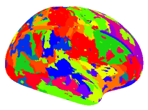
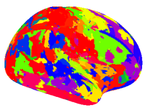
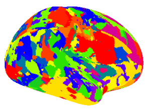
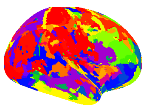
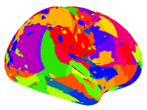
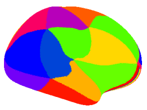
λ_2 λ_1	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48								
0.5								
0.5208								
0.5425								
0.5651								
0.5887								
0.6132								
0.6388								
0.6654								
0.6931								
0.722								

Table S.4: The clusterings, for the *right* hemisphere, generated by HP-CONCORD followed by the *persistent homology* method, at the tuning parameter values: $\varepsilon = 0$ (generally corresponding to *more* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. Table S.12 presents the Jaccard scores (S.3) for these clusterings. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

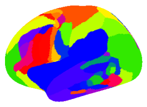
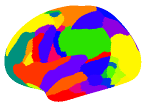
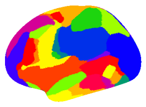
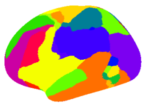
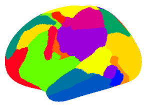
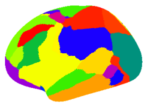
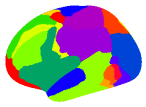
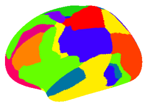
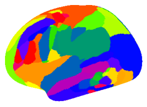
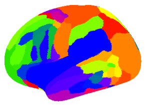
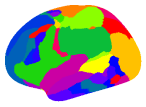
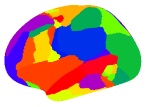
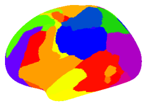
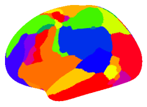
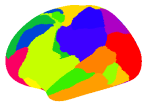
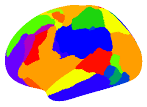
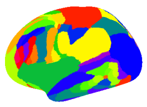
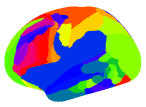
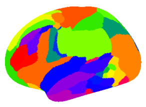
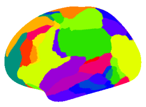
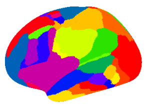
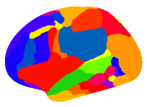
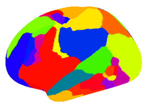
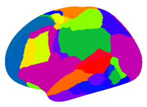
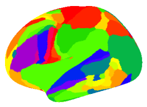
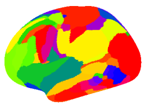
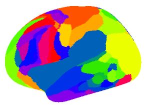
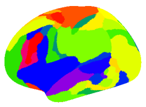
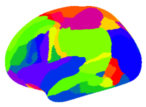
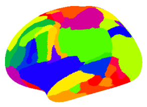
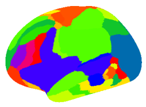
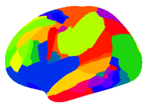
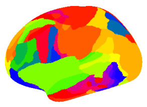
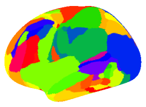
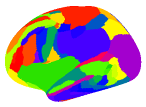
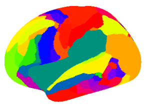
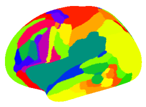
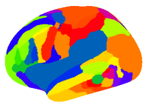
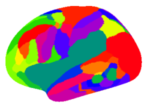
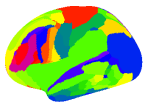
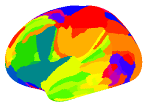
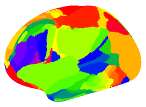
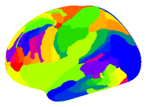
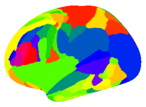
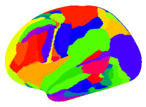
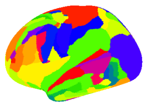
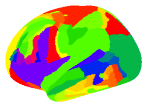
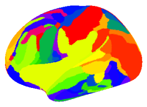
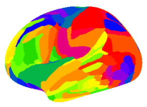
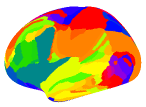
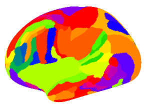
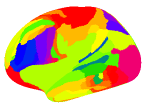
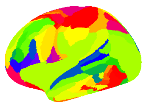
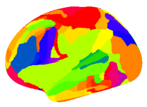
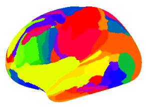
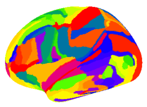
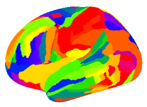
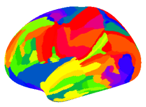
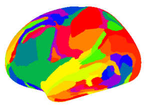
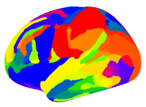
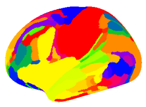
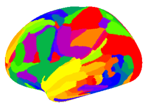
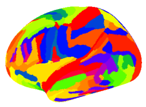
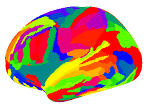
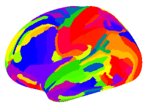
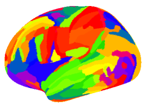
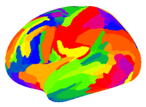
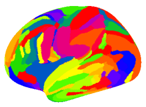
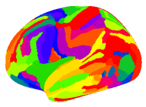
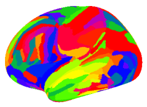
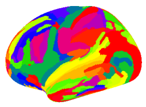
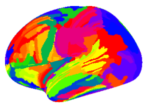
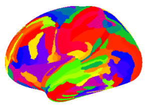
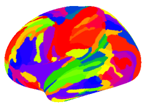
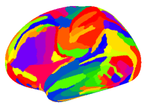
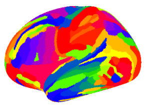
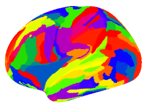
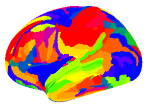
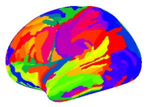
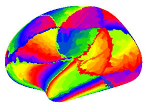
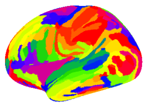
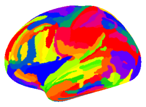
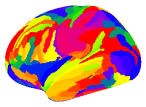
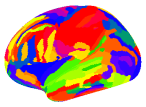
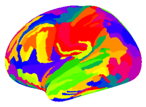
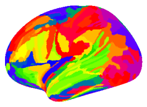
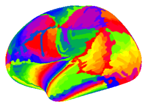
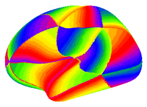
λ_1 \ λ_2	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48								
0.5								
0.5208								
0.5425								
0.5651								
0.5887								
0.6132								
0.6388								
0.6654								
0.6931								
0.722								

Table S.5: The clusterings, for the *left* hemisphere, generated by HP-CONCORD followed by the *Louvain* method, at the tuning parameter values: $k = 0$ (generally corresponding to *fewer* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. Table S.13 presents the Jaccard scores (S.3) for these clusterings. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

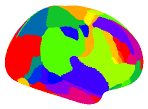
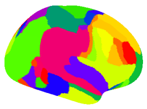
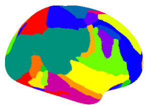
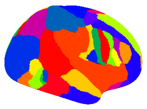
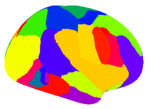
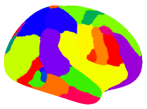
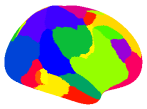
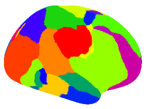
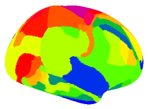
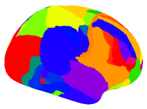
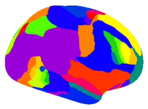
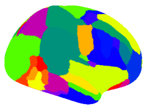
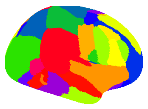
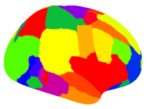
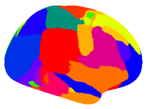
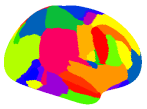
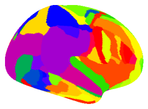
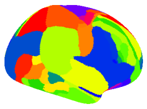
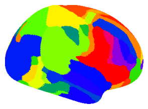
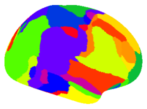
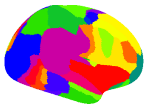
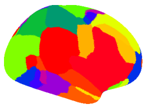
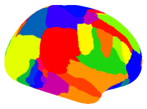
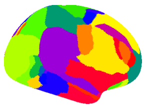
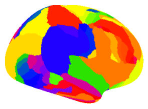
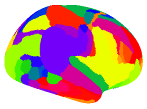
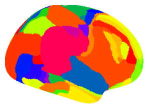
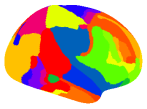
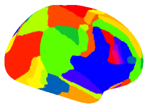
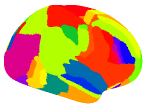
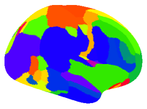
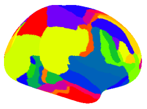
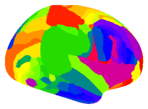
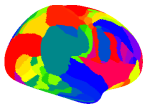
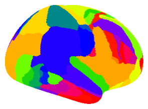
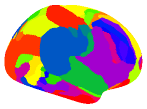
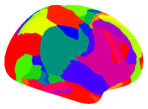
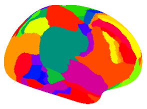
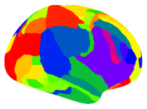
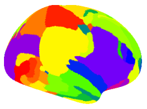
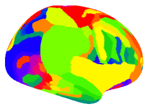
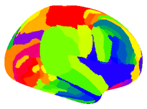
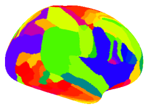
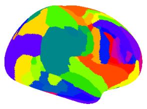
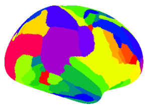
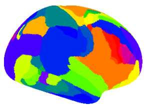
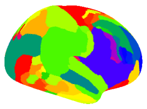
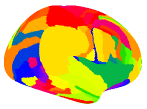
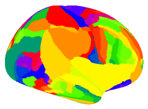
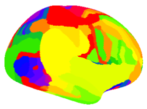
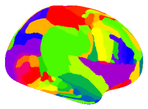
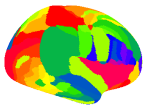
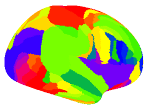
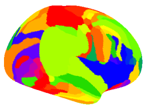
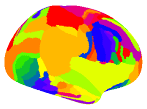
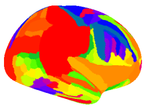
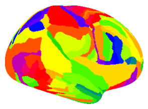
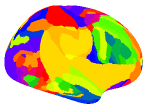
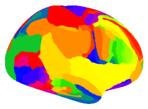
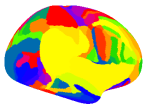
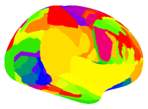
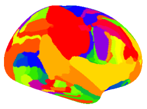
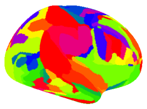
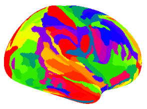
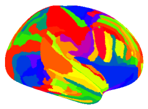
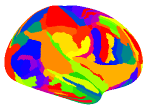
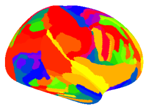
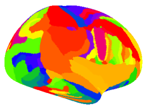
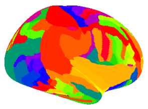
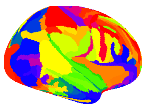
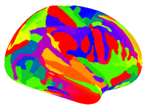
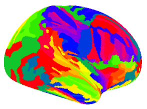
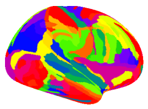
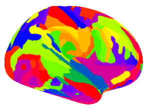
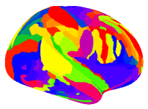
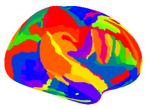
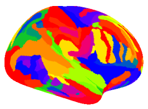
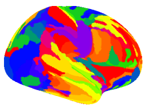
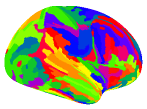
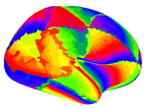
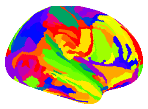
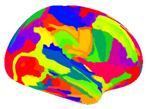
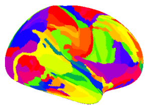
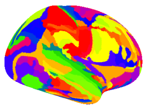
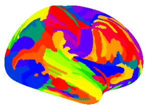
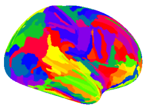
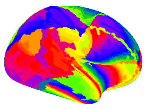
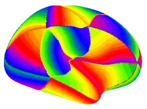
λ_1 \ λ_2	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48								
0.5								
0.5208								
0.5425								
0.5651								
0.5887								
0.6132								
0.6388								
0.6654								
0.6931								
0.722								

Table S.6: The clusterings, for the *right* hemisphere, generated by HP-CONCORD followed by the *Louvain* method, at the tuning parameter values: $k = 0$ (generally corresponding to *fewer* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. Table S.14 presents the Jaccard scores (S.3) for these clusterings. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

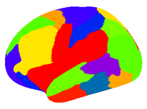
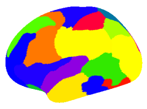
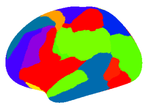
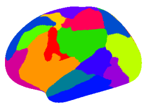
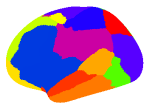
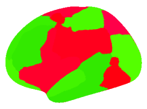
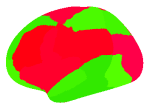
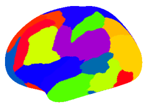
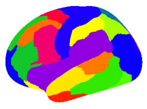
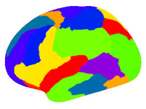
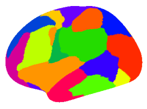
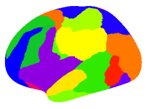
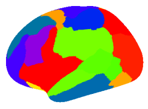
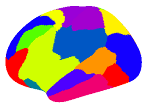
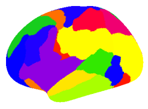
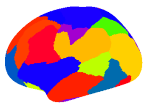
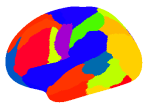
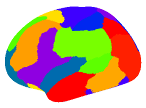
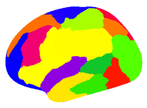
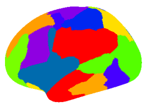
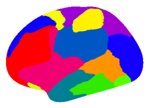
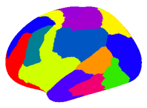
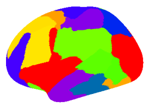
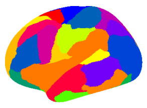
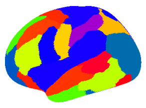
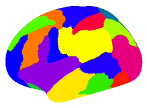
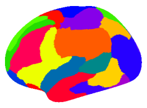
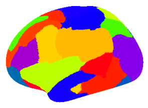
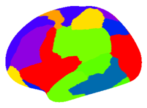
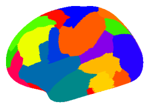
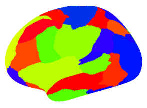
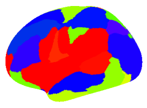
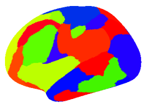
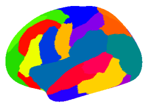
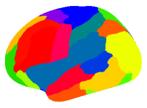
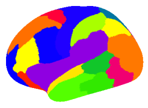
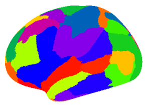
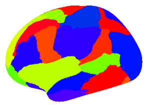
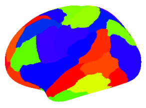
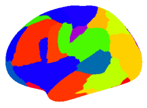
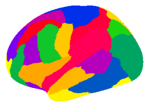
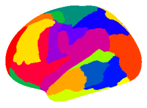
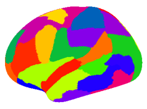
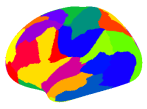
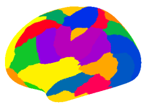
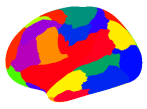
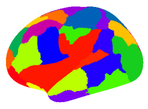
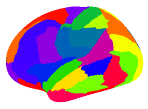
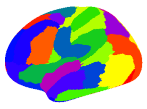
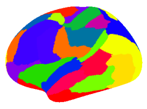
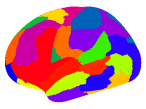
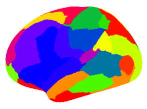
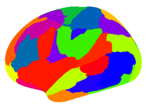
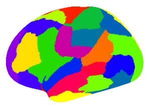
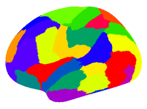
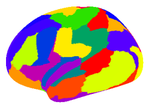
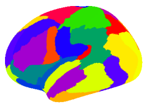
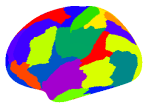
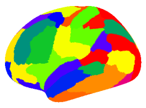
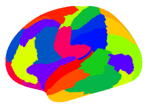
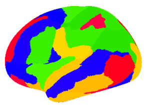
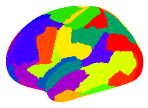
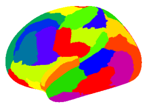
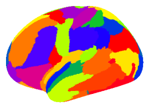
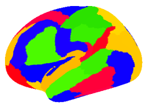
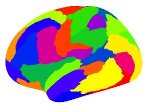
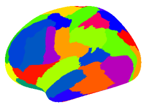
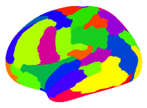
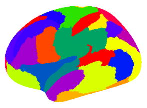
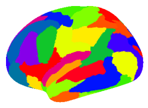
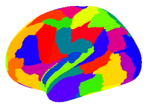
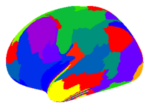
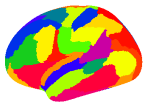
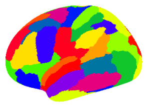
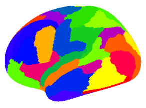
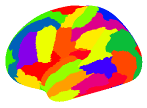
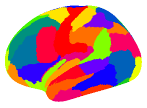
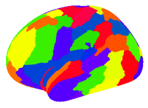
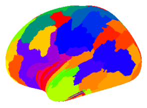
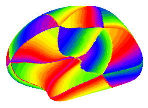
λ_2 λ_1	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48								
0.5								
0.5208								
0.5425								
0.5651								
0.5887								
0.6132								
0.6388								
0.6654								
0.6931								
0.722								

Table S.7: The clusterings, for the *left* hemisphere, generated by HP-CONCORD followed by the *Louvain* method, at the tuning parameter values: the largest value of k considered by Louvain (generally corresponding to *more* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. Table S.15 presents the Jaccard scores (S.3) for these clusterings. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

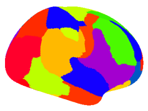
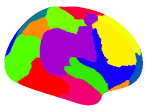
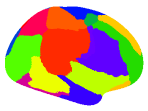
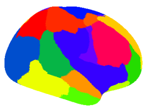
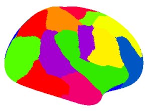
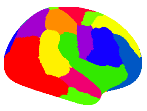
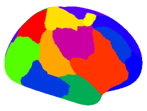
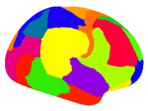
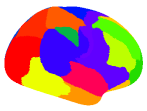
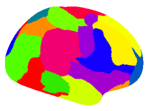
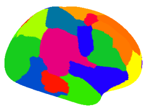
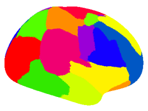
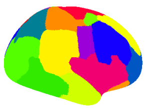
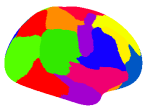
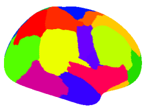
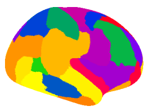
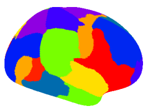
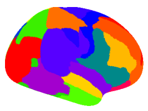
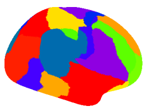
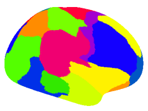
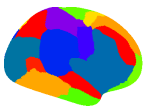
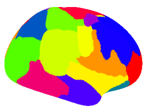
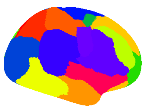
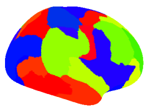
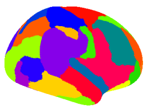
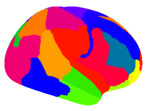
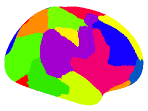
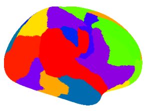
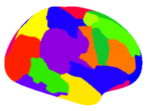
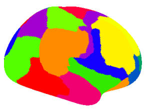
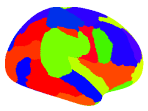
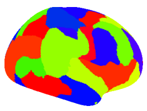
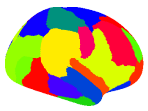
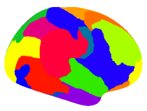
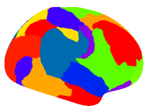
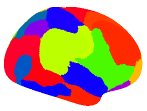
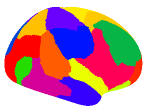
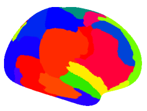
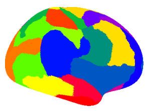
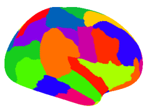
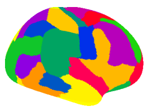
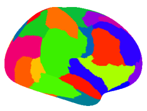
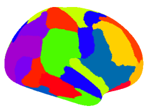
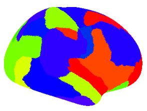
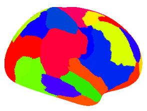
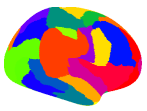
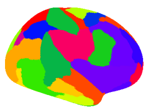
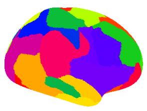
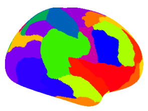
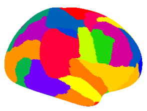
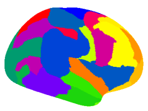
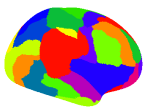
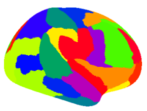
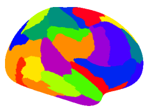
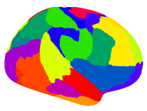
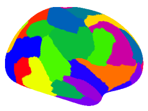
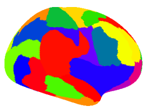
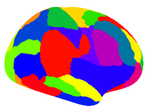
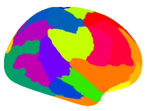
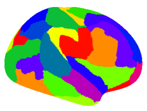
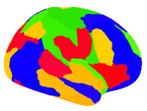
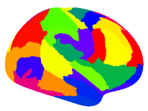
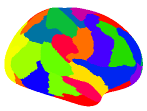
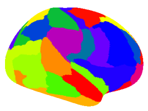
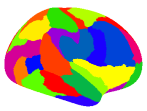
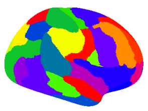
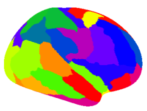
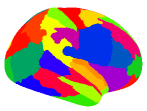
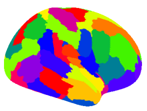
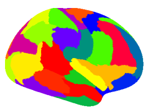
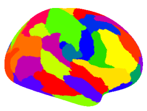
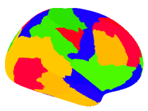
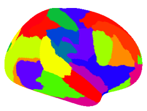
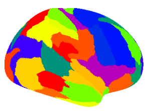
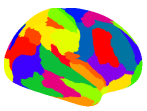
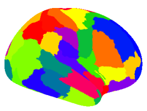
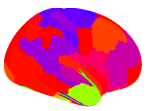
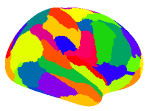
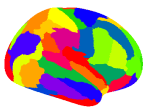
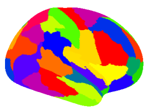
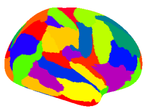
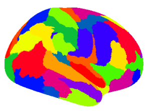
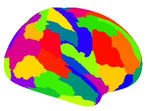
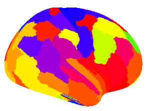
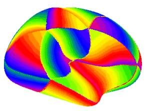
λ_2 / λ_1	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48								
0.5								
0.5208								
0.5425								
0.5651								
0.5887								
0.6132								
0.6388								
0.6654								
0.6931								
0.722								

Table S.8: The clusterings, for the *right* hemisphere, generated by HP-CONCORD followed by the *Louvain* method, at the tuning parameter values: the largest value of k considered by Louvain (generally corresponding to *more* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. Table S.16 presents the Jaccard scores (S.3) for these clusterings. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

λ_1 \ λ_2	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48	0.2043	0.2199	0.2242	0.2326	0.2277	0.2422	0.2447	0.23
0.5	0.2112	0.224	0.2315	0.2329	0.2343	0.2283	0.2197	0.2185
0.5208	0.1964	0.1895	0.2264	0.2385	0.2317	0.2282	0.2348	0.2358
0.5425	0.1905	0.1972	0.1951	0.2181	0.2268	0.2295	0.2289	0.2255
0.5651	0.1833	0.197	0.1973	0.1981	0.2125	0.2268	0.2242	0.2213
0.5887	0.1838	0.1845	0.1992	0.2067	0.1953	0.2057	0.2078	0.2155
0.6132	0.1702	0.1752	0.198	0.1995	0.2121	0.2014	0.2036	0.1891
0.6388	0.1698	0.1693	0.1864	0.1837	0.1859	0.191	0.1831	0.1785
0.6654	0.1538	0.1854	0.1759	0.1701	0.1748	0.1844	0.1805	0.1467
0.6931	0.1652	0.1689	0.1664	0.1686	0.1722	0.162	0.1472	0.0516
0.722	0.1382	0.1536	0.1556	0.1536	0.1442	0.1394	0.0758	—

Table S.9: The Jaccard scores (S.3) for the clusterings of the *left* hemisphere in Table S.1, generated by HP-CONCORD followed by the *persistent homology* method, at the tuning parameter values: $\varepsilon = 3$ (generally corresponding to *fewer* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

λ_1 \ λ_2	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48	0.2258	0.2315	0.2461	0.2279	0.2451	0.2436	0.2311	0.2431
0.5	0.2036	0.2245	0.2328	0.2326	0.2427	0.2314	0.2654	0.2528
0.5208	0.2255	0.2166	0.2317	0.2311	0.2427	0.2399	0.2381	0.2417
0.5425	0.21	0.2172	0.232	0.2355	0.2279	0.2299	0.245	0.2349
0.5651	0.2233	0.2182	0.2236	0.2341	0.2367	0.231	0.2286	0.2413
0.5887	0.2055	0.2187	0.2179	0.2369	0.2261	0.2321	0.2279	0.2067
0.6132	0.1843	0.2002	0.2245	0.2224	0.2113	0.219	0.2256	0.21
0.6388	0.1817	0.1843	0.2024	0.204	0.2154	0.2161	0.1981	0.1826
0.6654	0.1786	0.1678	0.1824	0.1891	0.1952	0.1749	0.1851	0.1273
0.6931	0.1652	0.1714	0.1686	0.1736	0.1714	0.1702	0.1284	0.061
0.722	0.1372	0.1562	0.162	0.1563	0.1364	0.1264	0.0875	—

Table S.10: The Jaccard scores (S.3) for the clusterings of the *right* hemisphere, generated by HP-CONCORD followed by the *persistent homology* method, at the tuning parameter values: $\varepsilon = 3$ (generally corresponding to *fewer* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

$\lambda_1 \backslash \lambda_2$	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48	0.0507	0.051	0.0527	0.0532	0.0511	0.0503	0.051	0.0518
0.5	0.053	0.0518	0.052	0.0519	0.0526	0.0531	0.0517	0.0516
0.5208	0.0517	0.0519	0.0527	0.0517	0.0524	0.0526	0.0522	0.0536
0.5425	0.0522	0.0519	0.0509	0.0516	0.0516	0.0514	0.0532	0.0533
0.5651	0.0514	0.0524	0.0512	0.0528	0.0529	0.0518	0.0518	0.0533
0.5887	0.0498	0.0524	0.0534	0.0521	0.0522	0.0521	0.0526	0.0532
0.6132	0.0504	0.0501	0.0531	0.052	0.0523	0.0529	0.0523	0.0505
0.6388	0.053	0.052	0.0494	0.0502	0.0517	0.0502	0.0516	0.0543
0.6654	0.0526	0.0529	0.0536	0.0533	0.0537	0.0506	0.0535	0.0558
0.6931	0.0529	0.054	0.0518	0.052	0.0532	0.0543	0.0566	0.0815
0.722	0.0549	0.0528	0.0525	0.0534	0.056	0.0561	0.0718	0.0884

Table S.11: The Jaccard scores (S.3) for the clusterings of the *left* hemisphere, generated by HP-CONCORD followed by the *persistent homology* method, at the tuning parameter values: $\varepsilon = 3$ (generally corresponding to *more* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

$\lambda_1 \backslash \lambda_2$	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48	0.0532	0.0506	0.0519	0.0516	0.0522	0.0521	0.0516	0.0508
0.5	0.0538	0.0536	0.0513	0.0512	0.0515	0.0525	0.0524	0.0506
0.5208	0.052	0.0519	0.0521	0.0509	0.0527	0.0517	0.0522	0.0509
0.5425	0.0505	0.0523	0.0528	0.0532	0.0511	0.0529	0.0526	0.0522
0.5651	0.0523	0.0501	0.0513	0.0513	0.053	0.0521	0.0512	0.0528
0.5887	0.0516	0.0528	0.0504	0.0515	0.0518	0.0515	0.0523	0.0511
0.6132	0.0505	0.0517	0.0525	0.0534	0.0511	0.0516	0.0543	0.0534
0.6388	0.0514	0.0543	0.0516	0.0522	0.0519	0.0533	0.0532	0.0544
0.6654	0.0544	0.0528	0.0514	0.0518	0.0525	0.0529	0.0565	0.061
0.6931	0.0527	0.0555	0.0525	0.0528	0.055	0.0529	0.0597	0.0845
0.722	0.0535	0.0524	0.0535	0.0527	0.0553	0.0566	0.0698	0.0918

Table S.12: The Jaccard scores (S.3) for the clusterings of the *right* hemisphere, generated by HP-CONCORD followed by the *persistent homology* method, at the tuning parameter values: $\varepsilon = 0$ (generally corresponding to *more* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

λ_1 \ λ_2	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48	0.1069	0.1101	0.0956	0.1042	0.1015	0.0958	0.0901	0.0905
0.5	0.1123	0.1076	0.1065	0.102	0.1089	0.1053	0.1007	0.107
0.5208	0.1097	0.111	0.1092	0.1096	0.1065	0.0982	0.1001	0.105
0.5425	0.1313	0.1123	0.1148	0.1085	0.1166	0.1143	0.1065	0.117
0.5651	0.1258	0.1216	0.1134	0.1167	0.1164	0.1097	0.1151	0.12
0.5887	0.129	0.1228	0.1233	0.1091	0.1203	0.1205	0.1238	0.1188
0.6132	0.1337	0.1294	0.1298	0.1289	0.1185	0.1285	0.1231	0.1455
0.6388	0.1477	0.1368	0.1363	0.1296	0.131	0.1344	0.1473	0.1517
0.6654	0.1486	0.1486	0.1458	0.1405	0.1488	0.1534	0.1486	0.1583
0.6931	0.1469	0.1453	0.1512	0.1483	0.146	0.1627	0.1706	0.0273
0.722	0.1581	0.1608	0.1557	0.1608	0.1661	0.1779	0.0461	0.0061

Table S.13: The Jaccard scores (S.3) for the clusterings of the *left* hemisphere, generated by HP-CONCORD followed by the *Louvain* method, at the tuning parameter values: $k = 0$ (generally corresponding to *fewer* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

λ_1 \ λ_2	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48	0.1105	0.1014	0.1034	0.1042	0.0988	0.0976	0.0976	0.0935
0.5	0.1084	0.1064	0.1011	0.1105	0.1003	0.1012	0.0992	0.1022
0.5208	0.1263	0.1059	0.1195	0.1056	0.0995	0.1059	0.1	0.1051
0.5425	0.122	0.1167	0.1113	0.1111	0.0997	0.1085	0.1125	0.0997
0.5651	0.1219	0.1212	0.1144	0.1022	0.1044	0.1109	0.1029	0.1189
0.5887	0.1218	0.1205	0.1184	0.1219	0.1159	0.1202	0.1183	0.135
0.6132	0.132	0.1259	0.1339	0.1265	0.1269	0.124	0.1294	0.1361
0.6388	0.1362	0.1364	0.1289	0.1286	0.1318	0.1279	0.1357	0.158
0.6654	0.1483	0.1451	0.1428	0.142	0.1438	0.1498	0.1626	0.1675
0.6931	0.1518	0.1552	0.1451	0.1473	0.1552	0.1671	0.1736	0.027
0.722	0.1648	0.1725	0.1556	0.1607	0.1643	0.1758	0.0482	0.0061

Table S.14: The Jaccard scores (S.3) for the clusterings of the *right* hemisphere, generated by HP-CONCORD followed by the *Louvain* method, at the tuning parameter values: $k = 0$ (generally corresponding to *fewer* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

λ_1 \ λ_2	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48	0.1678	0.1666	0.154	0.14	0.1297	0.135	0.1311	0.1284
0.5	0.1632	0.1778	0.1595	0.1515	0.1537	0.1454	0.128	0.1422
0.5208	0.1578	0.1719	0.1589	0.1663	0.1604	0.1572	0.145	0.1609
0.5425	0.1538	0.1572	0.166	0.1542	0.1702	0.1689	0.1684	0.1569
0.5651	0.1503	0.1602	0.1541	0.1473	0.1561	0.1587	0.1502	0.155
0.5887	0.1526	0.158	0.1537	0.1622	0.1564	0.1547	0.149	0.1338
0.6132	0.1438	0.1425	0.154	0.1487	0.151	0.1489	0.1327	0.1191
0.6388	0.1414	0.1453	0.134	0.1431	0.1393	0.1357	0.1238	0.0967
0.6654	0.1252	0.1263	0.1403	0.1301	0.1279	0.1196	0.0987	0.0653
0.6931	0.1137	0.1159	0.1161	0.1163	0.109	0.0937	0.0701	0.0216
0.722	0.1008	0.1005	0.1015	0.0961	0.0891	0.0679	0.0298	0.0061

Table S.15: The Jaccard scores (S.3) for the clusterings of the *left* hemisphere, generated by HP-CONCORD followed by the *Louvain* method, at the tuning parameter values: the largest value of k considered by Louvain (generally corresponding to *more* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

λ_1 \ λ_2	0.1024	0.128	0.16	0.2	0.25	0.3125	0.3906	0.4883
0.48	0.1719	0.1697	0.1633	0.1763	0.1499	0.1475	0.1442	0.1365
0.5	0.1689	0.1675	0.167	0.17	0.1661	0.1587	0.1411	0.1729
0.5208	0.1651	0.1581	0.1808	0.1694	0.1655	0.1528	0.153	0.1512
0.5425	0.1697	0.1556	0.1634	0.1637	0.1591	0.1581	0.1857	0.1598
0.5651	0.1651	0.1663	0.1509	0.1554	0.1567	0.1542	0.151	0.1416
0.5887	0.1492	0.1602	0.1635	0.1541	0.1512	0.1586	0.1506	0.1536
0.6132	0.1474	0.1596	0.1586	0.1593	0.1649	0.1548	0.1436	0.1168
0.6388	0.1321	0.1337	0.1495	0.1502	0.1458	0.1272	0.1188	0.0938
0.6654	0.119	0.1203	0.1233	0.1221	0.1185	0.1125	0.0973	0.0635
0.6931	0.112	0.1136	0.1128	0.111	0.107	0.0932	0.0694	0.0213
0.722	0.0943	0.098	0.0994	0.0937	0.0832	0.0672	0.0299	0.0061

Table S.16: The Jaccard scores (S.3) for the clusterings of the *right* hemisphere, generated by HP-CONCORD followed by the *Louvain* method, at the tuning parameter values: the largest value of k considered by Louvain (generally corresponding to *more* clusters) as well as all the λ_1, λ_2 values we describe in Section 5. "—", if present, indicates a degenerate clustering that puts either all the voxels into a single cluster or each voxel into its own cluster.

References

[1] A. Azad, J. Langguth, Y. Fang, A. Qi, and A. Pothen. Identifying rare cell populations in comparative flow cytometry. In *Proceedings of the 10th International Workshop on Algorithms in Bioinformatics*, 2010.

[2] E. Belilovsky, K. Kastner, G. Varoquaux, and M. Blaschko. Learning to discover graphical model structures. *arXiv:1605.06359*, 2016.

[3] M. Couprie and G. Bertrand. Topological gray-scale watershed transform. In *Proceedings of the International Symposium on Optical Science and Technology: Vision Geometry VI*, 1997.

[4] H. Edelsbrunner and D. Morozov. Persistent homology: theory and applications. In *Proceedings of the 6th European Congress of Mathematics*, 2012.

[5] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

[6] Z. Fu, S. Han, A. Tan, Y. Tu, and Z. Zhang. ℓ_0-regularized time-varying sparse inverse covariance estimation for tracking dynamic fMRI brain networks. In *Proceedings of the 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2015.

[7] J. Peng, P. Wang, N. Zhou, and J. Zhu. Partial correlation estimation by joint sparse regression models. *Journal of the American Statistical Association*, 104(486):735–746, 2009.

[8] J. Ramsey, M. Glymour, R. Sanchez-Romero, and C. Glymour. A million variables and more: the fast greedy equivalence search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images. *International Journal of Data Science and Analytics*, 3(2):121–129, 2017.

[9] J. D. Ramsey. Scaling up greedy causal search for continuous variables. *arXiv:1507.07749*, 2015.

[10] S. Ryali, T. Chen, K. Supekar, and V. Menon. Estimation of functional connectivity in fMRI data using stability selection-based sparse partial correlation with elastic net penalty. *NeuroImage*, 59(4):3852–3861, 2012.

[11] S. M. Smith, C. F. Beckmann, J. Andersson, E. J. Auerbach, J. Bijsterbosch, G. Douaud, E. Duff, D. A. Feinberg, L. Griffanti, M. P. Harms, et al. Resting-state fMRI in the human connectome project. *NeuroImage*, 80:144–168, 2013.