

TbCAPs User Manual

Welcome to the TbCAPs User Manual, and to co-activation pattern (CAP) analysis! In what follows, you will be able to read about all that you need to get started with the toolbox, including some theoretical background, empirical advice based on our own experience with CAPs, and an exhaustive description of the available functionalities within the current interface. As another useful resource, we also point you at the companion manuscript to this User Manual (Bolton et al., 2019). We hope that you will enjoy the journey, and wish you a pleasant reading!

1. Overview of co-activation pattern analysis

You may wish to skip this section if you are already well acquainted with the theoretical aspects underlying co-activation pattern analysis. Figure 1 summarises its main points of interest; in panel A, you see a recent summary of what the so called *dynamic functional connectivity* (dFC) field is shaped like: from functional magnetic resonance imaging (fMRI) data (more precisely, the blood oxygenation level-dependent—BOLD—contrast), the goal is to track how different brain locations reconfigure their interactions over the course of time.

To date, this has been a particularly influential endeavour in the context of resting-state (RS) recordings, where the scanned subjects lie at rest in the scanner and are asked not to perform any externally-oriented, cognitively-demanding task. The rationale for using RS data is a shorter acquisition time, and the lack of any task instructions that should be followed by the subjects; as such, an extended range of clinical populations may be analysed (to the extent that data remains of sufficiently good quality, a point to which we will come back below). The analysis of functional brain dynamics at rest has already started to show its potential in refining our understanding of human cognition or disease.

In the left side of panel A, you can see how most analytical approaches to date operate: statistical interdependence (typically termed *functional connectivity*—FC) between brain regions is computed as Pearson’s correlation coefficient (or other statistical measures) within consecutive temporal sub-windows, and doing so between all possible pairs of regions enables to generate a temporal series of FC matrices. These may be analysed as such—for instance by computing the standard deviation of a dFC time course over time, or decomposed into a discrete set of recurring dynamic functional connectivity states. For more extensive information about this wide field of research, we invite you to have a read at a few contemporary dFC reviews (Hutchison et al., 2013; Preti et al., 2017; Lurie et al., 2018).

Co-activation pattern analysis follows another conceptual track, as outlined in red in panel A: instead of computing second-order statistics, the analysis operates at the frame-wise level, directly from the voxel-wise BOLD time courses. There are two main stages in the analysis: first, a subset of time points when a seed region of interest becomes strongly (de)active are selected (as denoted by the top right sub-panel). Second, these fMRI volumes—for which the seed at hand is always active—are separated into a set of whole-brain co-activation patterns (that is, sets of regions that co-(de)activate with the seed at different time points).

In panel B, we first focus on the selection of relevant time points to the analysis. On the far left, the spatial maps resulting from an average of retained fMRI frames using different seeds (labeled “Condition rates”) are compared to the outputs from an Independent Component Analysis (ICA) (labeled ‘PICA’), for six different large-scale RS networks. You can see that this time point selection approach enables to retrieve spatially similar maps compared to ICA, and this holds true when averaging together as little as 3 distinct frames only!

The more data points are averaged together (that is, the lower the threshold past which seed activity is deemed significant), the closer the resulting average map gets to the corresponding ICA map.

On the right hand side, we see a similar comparative analysis with the outputs from a seed-based correlation analysis, where the seed region’s averaged BOLD time course is correlated with all other voxels from the brain. In the case of a posterior cingulate cortex (PCC) seed—a member of the default mode network (DMN)—we can already see the DMN from single frames extracted through the seed thresholding process, and the DMN pattern gets clearer and clearer as we move to an average over all retained frames within a single run, or within an extended set of runs. Furthermore, a strong similarity to the seed-based correlation map is observed, even if as little as 10% of frames are used in the averaging process.

To summarise until now, the frames that are selected in CAP analysis contain the information that is rendered by more classical approaches such as ICA or seed-based correlation. Furthermore, in the perspective of resorting to very large-sized datasets, it is very desirable to analyse only a tenth or so of the available data, given the lowered computational load!

Panel C summarises why CAP analysis actually goes beyond the aforementioned, classical analytical tools: as illustrated in the middle left, the retained frames across subjects are actually not averaged, but instead, *clustered* through k-means clustering (for details on this widely applied unsupervised approach, see Shukla and Naganna (2014) or https://en.wikipedia.org/wiki/K-means_clustering). Thus, instead of obtaining one “average map” of seed co-(de)activation, we generate a set of K different such instances: the underlying hypothesis is that *at different time points, the seed region will co-(de)activate with different sets of areas*.

Over the past years, it has become clear that the temporal expression of CAPs is not random, but follows clear rules: for instance, the bottom left of panel C shows how different CAPs in the mouse are expressed at specific moments within the temporal oscillations taken by the global signal (average BOLD signal throughout the brain). To capture such temporal relationships, a wide array of metrics are computed by our toolbox, including some reflective of CAP-to-CAP transitions.

In addition, the right hand side of panel C shows that CAPs also contain precious spatial information: in that particular case, $K = 8$ CAPs were extracted from the signal of a PCC seed, and thus stand for DMN sub-components. Compared to the average map (at the top), each CAP also contains specific, additional foci of (de)activation, and thus, CAP analysis offers a window on how the exact spatial patterns of resting-state network expression change over time.

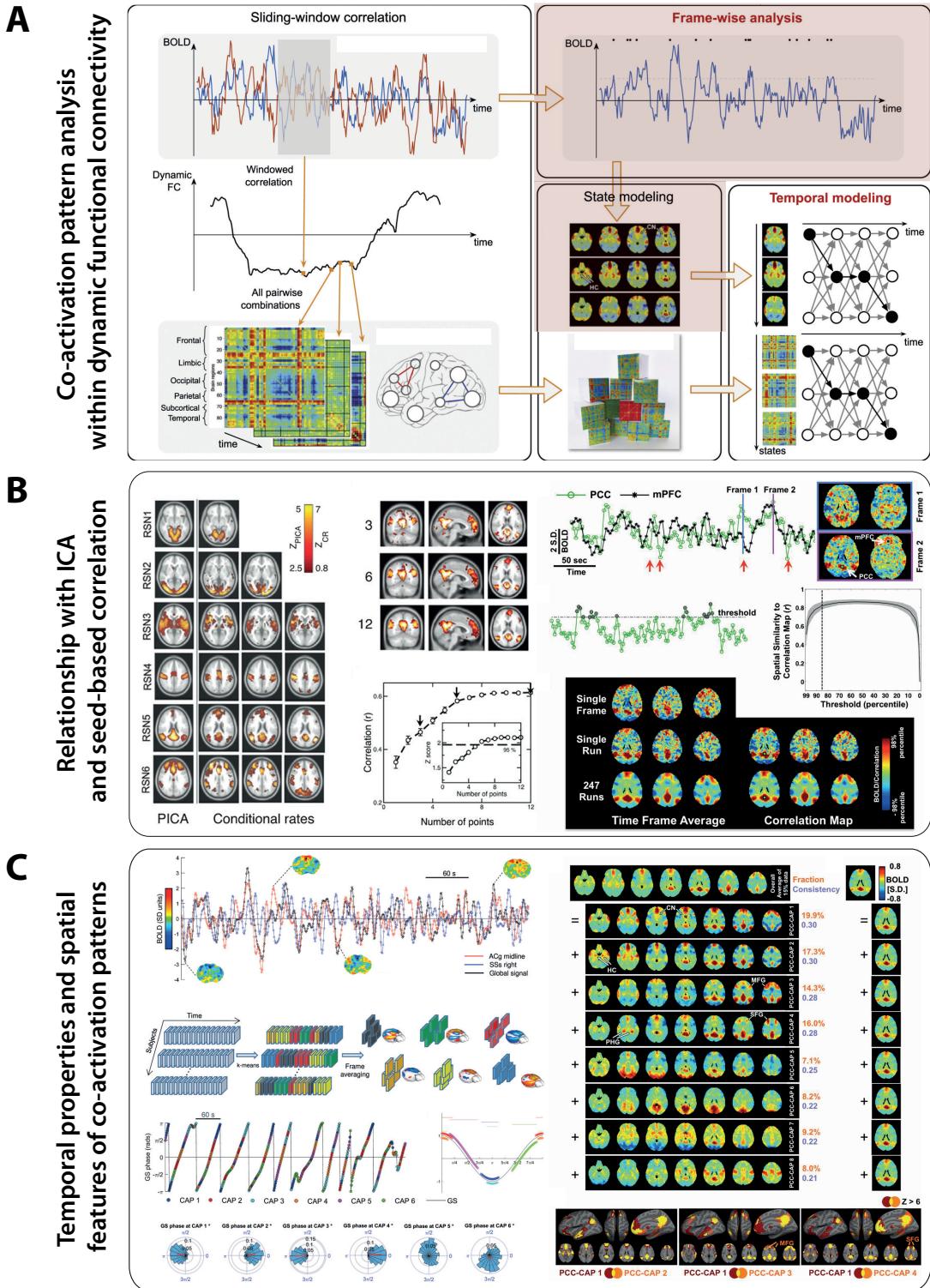


Figure 1: Overview of co-activation pattern analysis (A) Co-activation pattern analysis is located within the landscape of dynamic functional connectivity approaches (red shaded sub-panels). Figure reproduced from Preti et al. (2017). (B) The average of selected frames within co-activation pattern analysis shows strong similarity to Independent Component Analysis (left half) and seed-based correlation (right) maps. The panel halves are reproduced from Tagliazucchi et al. (2012) (left) and Liu and Duyn (2013) (right). (C) When selected frames are clustered instead of averaged, co-activation patterns are generated. The expression of co-activation patterns obeys a precise temporality (bottom left), and each map displays subtle spatial sub-components that are not seen with a mere averaging, and more accurately render the expression of resting-state networks. The panel halves are reproduced from Gutierrez-Barragan et al. (2019) (left) and Liu and Duyn (2013) (right).

2. Guided TbCAPs example

2.1. Data preparation

In what follows, we will be guiding you through an example of use of TbCAPs, so that you know how to prepare your data and what to expect. In this simple example, we will be analysing the data from 15 healthy controls, acquired at a TR of 3.3 s. **Preprocessing of the data should be done prior to the use of the toolbox.** The steps that we judge compulsory as part of a viable preprocessing¹ are, for each run to analyse: (1) realignment of the functional volumes (by which estimates of motion time courses are generated), (2) co-registration of the functional data to the structural T1 volume, (3) linear detrending and regression of covariates of no interest from the BOLD voxelwise time courses (this can for example include constant, linear and quadratic trends, average white matter/cerebrospinal fluid time courses, motion time courses and their derivatives or squared expansions), (4) filtering of the BOLD time courses (we recommend high-pass—but not low-pass—filtering at around 0.01 Hz), (5) nonlinear warping into MNI space, and (6) smoothing (we recommend 5 mm full width at half maximum). Of course, the exact implementation of each step may vary from case to case (for example, filtering could be part of the regression process using a Discrete Cosine Transform basis). Note that you should not perform scrubbing (Power et al., 2012) yourself, as this is handled within the toolbox. The most widely used software to preprocess fMRI data are FSL (Jenkinson et al., 2012) or SPM (<https://www.fil.ion.ucl.ac.uk/spm/software/spm12/>), and there exist several toolboxes to further facilitate all steps (see for instance Yan and Zang (2010)).

Once the data has been preprocessed, **it should be arranged so that each run to analyse lies in a different directory**, as shown for our toy example in Figure 2. In this example case, all anatomical files lie in the “anat” folders (they will actually not be required in the analysis), and all functional files in the “func” folders. You should ensure the following for the toolbox to run properly: (1) the directories that contain the functional volumes of interest should be named similarly (*e.g.*, “func” in this case), (2) while you may have more than only the fully preprocessed fMRI volumes in these directories, they should be distinguishable on the basis of their prefix (in this example, “sw”, which denotes that the data has been warped and smoothed), (3) there should be the same number of volumes across all the runs to jointly analyse, and the volumes should all have the same spatial resolution², (4) the functional volumes should not be in 4D format, but present as separate 3D files with NIFTI format, and (5) in each of the directories containing the data to analyse, you should also add the text file summarising motion parameters (as the toolbox will use it to perform scrubbing of the data), with rotational parameters standing as the 4th to 6th columns, in radians; if this file is not present, the toolbox will assume zero movement, and scrubbing will not be performed.

¹Note that additional steps are strongly advised for optimal data preparation, as we touch upon below.

²Remember that all the data should be warped to MNI space, as we will be running population-level assessments!

Name		Date Modified	Size	Kind
▼ sub-01			--	Folder
► anat		Today, 15:00	--	Folder
► func		Today, 15:00	--	Folder
▼ sub-02		13 Jan 2020, 02:50	--	Folder
► anat		Today, 15:00	--	Folder
► func		Today, 15:00	--	Folder
▼ sub-03		13 Jan 2020, 02:50	--	Folder
► anat		Today, 15:00	--	Folder
▼ func		Today, 15:00	--	Folder
● swrfMRI_00001.nii		13 Jan 2020, 02:50	--	Folder
● swrfMRI_00002.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00003.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00004.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00005.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00006.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00007.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00008.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00009.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00010.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00011.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00012.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00013.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00014.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00015.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00016.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00017.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00018.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00019.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00020.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00021.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI
● swrfMRI_00022.nii		11 Oct 2017, 16:20	1.2 MB	NIFTI

Figure 2: How to arrange the data to analyse with TbCAPs.

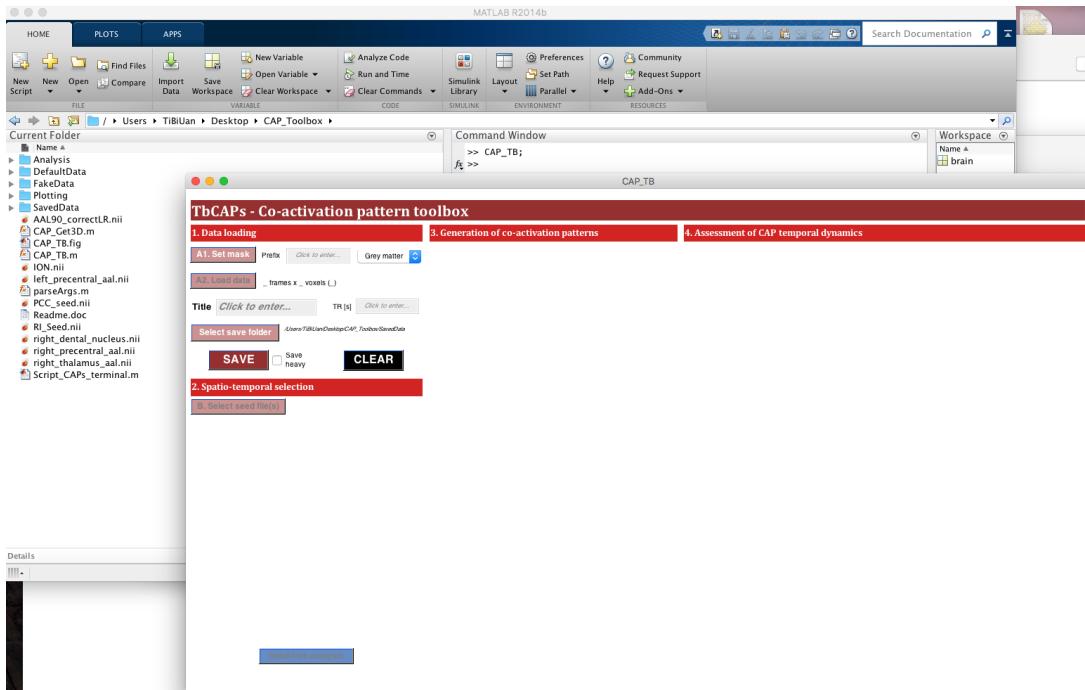


Figure 3: The starting TbCAPs interface.

2.2. Toolbox opening and data loading

Now that the data is ready, we can open the toolbox from the MATLAB terminal: note that since some SPM functions are used within TbCAPs, **SPM should have been previously installed, and added to the MATLAB path**. To open the toolbox Graphical User Interface (GUI), type “CAP_TB;” in the terminal; as a result, you will see the interface displayed in Figure 3 pop up.

For now, only the data loading part of the interface is enabled, and we should provide information regarding (1) the prefix that specifies the volumes to load (in the dedicated text box; for us, “sw”), and (2) the mask that we wish to use for the analyses. What we mean by the latter is that CAP analysis will not operate on all the voxels available in the fMRI volumes of interest: this would be computationally overly demanding, and useless since all out-of-brain voxels have an uninformative value of 0 anyway. From our past experiences, **we strongly advise to use as stringent a mask as possible**, both for the sake of computational speed and because otherwise, “noisy” voxels that remain may bias the results of the analyses (for example, some cerebrospinal fluid voxels could occasionally take very large BOLD signal values due to physiological influences, and alter the k-means clustering process). The default prefix assumed by the toolbox if nothing is entered is “sw”, and the default mask is a grey matter one, which we encourage you to select.

After entering this information, clicking on the **A1. Set mask** button prompts us to select one of the directories containing our functional data (see Figure 4); this is so that the originally high resolution mask can be converted to the resolution of the functional data, as it becomes known by the toolbox. If this step ends properly, we are then allowed to click on the **A2. Load data** button, where we are prompted to select all the directories where functional volumes of interest lie (Figure 5): there should be one such directory per run

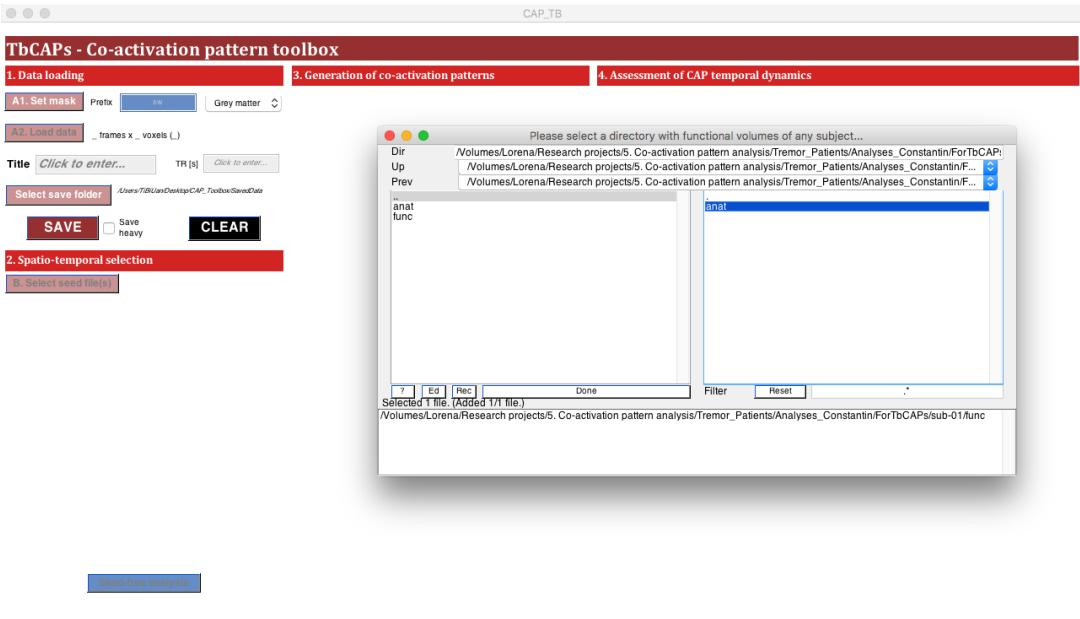


Figure 4: Preparing the mask for CAP analysis.

to consider. Selected data will be gradually loaded into the toolbox (note that it will also undergo temporal z-scoring at this time, a required step in CAP analysis), and when successfully achieved, the spatio-temporal selection utilities appear on screen. In addition, the dimensionality of your data (number of frames and of voxels input to the analysis) will be summarised to the right of the **A2. Load data** button.

Note that you may enter other information of interest, such as the TR of your data (this will be used to adjust some of the axes of plots that will be generated later), the folder where you wish to save the results of your analyses (by default, the “SavedData” folder present in the downloadable toolbox version), and a name for your project that will help you fetch that saved data. You may save data at any stage, but right now, it would be completely useless!

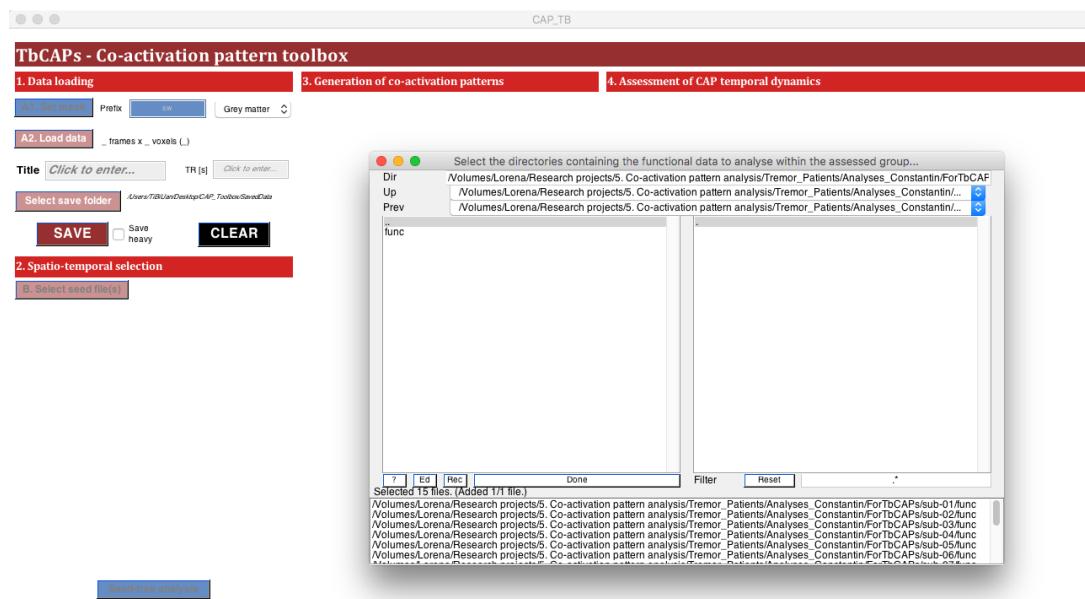


Figure 5: Loading the data to analyse.

2.3. Spatio-temporal selection

In this phase, we want to specify all the details regarding the selection of time points for CAP analysis. The first step is of course to provide the toolbox with information regarding the seed(s) at hand: in our running example, we will only load one seed, but up to three may be loaded jointly (see below for more details). The seed file should be in NIFTI format and MNI space, but it can be of any spatial resolution (the toolbox will convert it automatically to that of the functional data). To load it, we should click on the **B. Select seed file(s)** button, and then select the NIFTI file of interest³. In this example, we use the bilateral posterior cingulate cortex from the AAL atlas (Tzourio-Mazoyer et al., 2002).

When successfully loaded, parameters associated to this analytical stage become visible, and we can also plot brain slices enabling us to verify the seed location, and, as a sanity check, the average seed-based correlation map (in our case, it expectedly reflects the classical DMN). All these can be navigated through thanks to dedicated sliders. Thanks to the “+” and “-” buttons on the left hand side, we can decide whether we want to analyse, respectively, the frames showing strong seed activation or deactivation. Within the “Selection” panel, we can also decide whether we want to select the time points of interest using a threshold T (in which case, there may be different amounts of retained frames across analysed runs), or specifying a fixed percentage P . In addition, we should decide at which cutoff framewise displacement value M the data will be censored. In our running example, we will analyse episodes of seed activation, using a threshold $T = 1$ and scrubbing at $M = 0.5$ mm. Since the seed time course is z-scored (that is, has a mean of 0 and a standard deviation of 1), if the data were normally distributed, we would retain 16% of time points at $T = 1$. We can see that the actual amount of retained frames, displayed in the bottom right violin plot/box plot representation, is fairly compatible with this reasoning. Note that this plot considers the data following scrubbing. Figure 6 shows the current status of the TbCAPs window at this stage, where it can be seen that the generation of co-activation patterns is now enabled.

³Alternatively, it is also possible to run a seed-free analysis if clicking on the **Seed-free analysis** button; we come back to that functionality later.

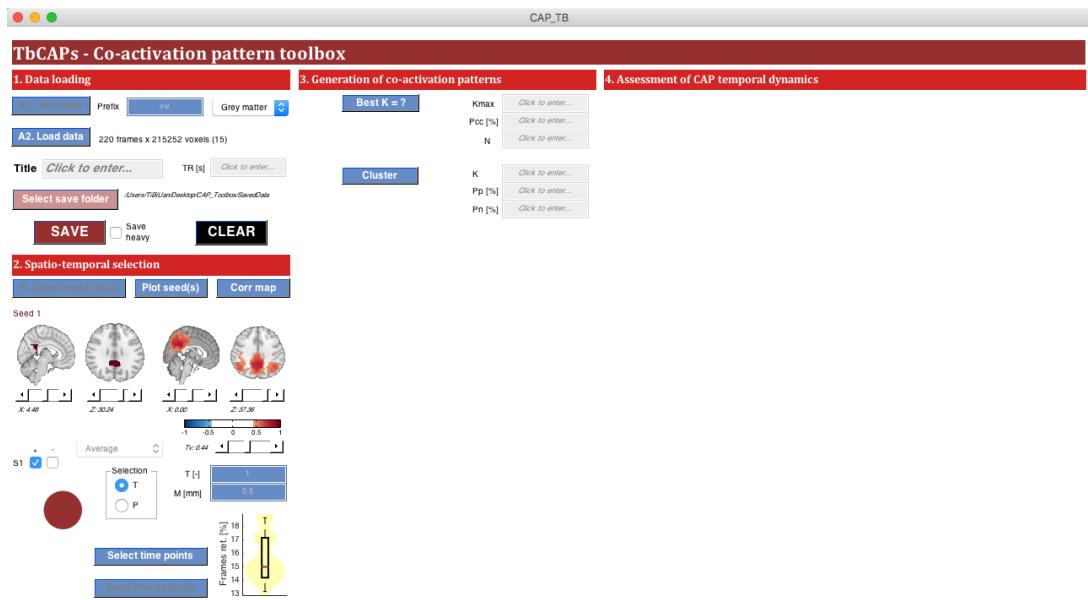


Figure 6: State of the interface following spatio-temporal selection.

2.4. Generation of co-activation patterns

To generate CAPs, we should specify the number of clusters K into which to partition the data. In most occasions, this value is not known *a priori*, so we wish to resort to data-driven methods that enable us to pick an optimum. Several methods exist for this purpose, including for example the use of Silhouettes (Rousseeuw, 1987) or the location of an *elbow* in the quality metric curve (Kodinariya and Makwana, 2013). Here, we implemented an approach termed *consensus clustering*, introduced by Monti et al. (2003) and deemed particularly effective in the case of dimensionally large datasets. Accordingly, recent neuroimaging work has seen the successful application of this technique to select K in the generation of large-scale functional brain networks (Zöller et al., 2018; Bolton et al., 2019). In the following paragraphs, we briefly elaborate on the mathematical underpinnings of consensus clustering.

Let us consider a candidate cluster number k , and two data points i and j (that is, two different retained fMRI volumes); k-means clustering is run over N folds, each time on a randomly selected subsample of $P_{CC}\%$ of the data, without replacement. For each fold f , we determine whether the data points are assigned to the same cluster, or to a different one: respectively, we then have $[\mathbf{C}_k^{(f)}]_{i,j} = [\mathbf{C}_k^{(f)}]_{j,i} = 1$ or 0, where the matrix $\mathbf{C}_k^{(f)}$ summarises the assignments for fold f and cluster number k .

The *consensus matrix* entry $[\mathbf{C}_k]_{i,j}$ for a given number of clusters k and two data points i and j can then be extracted by averaging over all folds where both data points jointly entered the computations (that is, were part of the sub-fraction of selected data). Let the set of such folds be denoted by \mathcal{F} , we then have:

$$[\mathbf{C}_k]_{i,j} = \frac{\sum_{f \in \mathcal{F}} [\mathbf{C}_k^{(f)}]_{i,j}}{|\mathcal{F}|}. \quad (1)$$

If the assessed cluster number is good, two given data points should consistently be either clustered together, or clustered separately, across folds; hence, $[\mathbf{C}_k]_{i,j}$ should be close to 1 or to 0, respectively. Intermediate values, however, would indicate inconsistent clustering across folds, which is undesirable. To quantify this, we first compute the cumulative distribution of consensus values across all pairs of data points (*i.e.*, sample the upper diagonal of the consensus matrix). We denote this distribution by $\mathcal{P}_k(c)$, with $c \in [0, 1]$ the consensus. We then compute the proportion of ambiguously clustered pairs—PAC (Senbabaoglu et al., 2014): let a threshold consensus value c_T above which we judge that assignment is not sufficiently homogeneous across folds, PAC for cluster number k is then given as:

$$\text{PAC}_k = \sum_{c=c_T}^{1-c_T} \mathcal{P}_k(c). \quad (2)$$

A lower PAC value highlights more robust clustering across folds. Results can be extracted for a range of candidate values (in the toolbox, K_{\max} specifies the maximal number of clusters to assess), and analysed for several threshold consensus values c_T , to determine optimal cluster numbers. Consensus clustering can be started by clicking on the **Best K = ?** button; when it has finished running in the toolbox⁴, the range of threshold consensus values is represented by the colour coding from yellow to black, and the stability measure $1 - \text{PAC}$ is plotted as the y-axis (hence, more positive values highlight a more robust k). $1 - \text{PAC}$ often exhibits an exponential increase with larger values for k ; in this case, the optimum should be selected beyond this general trend.

⁴Beware that this is by far the most demanding step, computationally speaking!

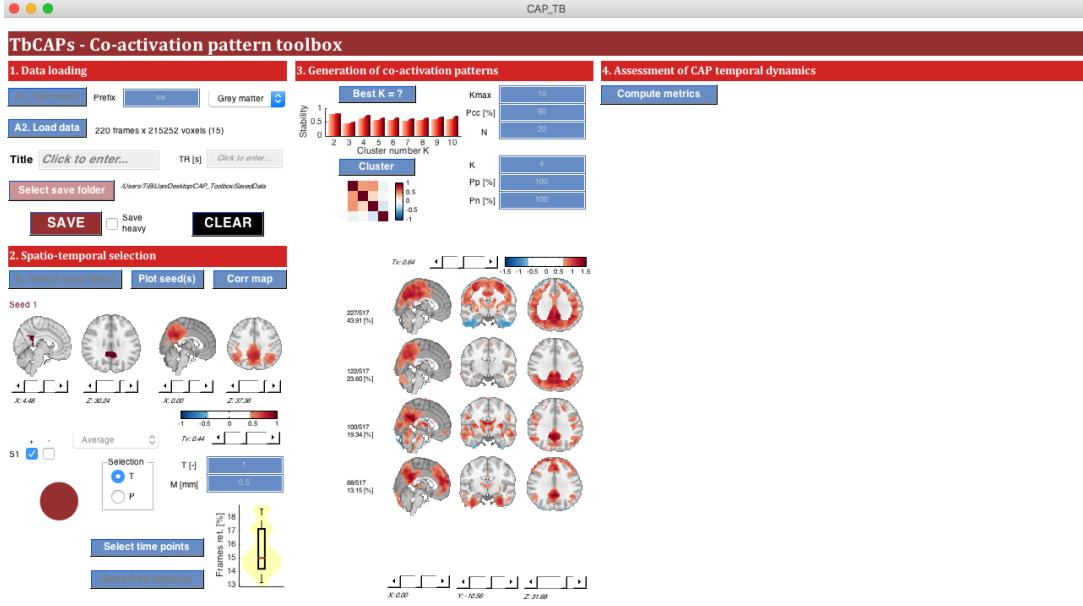


Figure 7: State of the interface following the generation of co-activation patterns.

In our running example, we will use $K = 4$ to generate CAPs, as it clearly stands out as a local stability optimum. We set both parameters P_P and P_N to 100%: they respectively specify the fraction of positive-valued and negative-valued voxels that should be kept for clustering (the rest is set to zero). These parameters are made available in the interface because past CAP works have sometimes relied on them, but we advise to cluster the whole pool of voxels without further tuning, so that no additional free parameter enters the analysis.

Clicking on the **Cluster** button starts the clustering process; when finished, the five most recurring CAPs in the subject population at hand become visible, and can be navigated through *via* their dedicated sliders. A matrix of spatial similarity across CAPs also becomes visible. Figure 7 shows the status of the interface at this stage; we can now move to the computation of temporal CAP metrics.

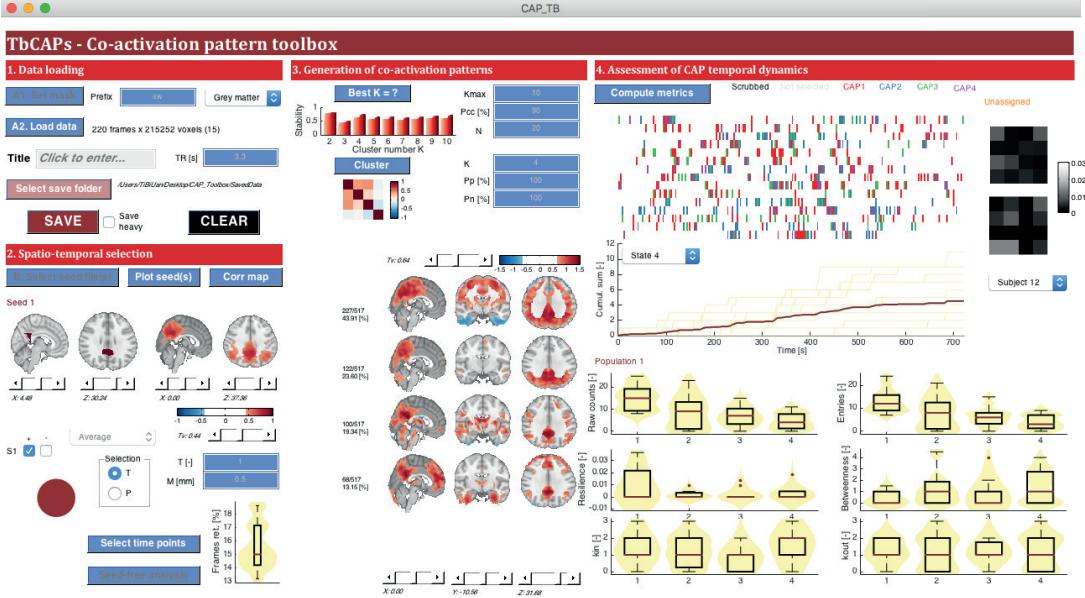


Figure 8: State of the interface following the computation of co-activation pattern metrics.

2.5. Assessment of CAP temporal dynamics

To compute CAP metrics, we only need to click on the **Compute metrics** button, following which all computed quantities become displayed on screen (Figure 8). At the top, we find a colour-coded display of which CAPs are expressed over time (from left to right) across the analysed subjects (from top to bottom). Just below this representation, we also have access to a cumulative plot showing the expression of a chosen CAP (in this illustration, CAP4, labeled as “State 4” in the dedicated popup menu), in individual subjects (thin lines) and as a population average (thick line). On the right hand side, the matrix of transition probabilities across all $K = 4$ CAPs is displayed, averaged across the population (top matrix), or for a selected individual subject (in this illustration, we selected “Subject 12” in the dedicated popup menu).

The six sets of violin plots/box plots at the bottom of the panel summarise the metric values across all subjects from the population, for raw occurrences (top left), number of entries (top right), resilience (*i.e.*, probability to remain in a given CAP from time t to $t + 1$; middle left), betweenness centrality (middle right), in-degree (that is, the likelihood to enter a CAP from any other), and out-degree (the likelihood to exit a CAP towards any other).

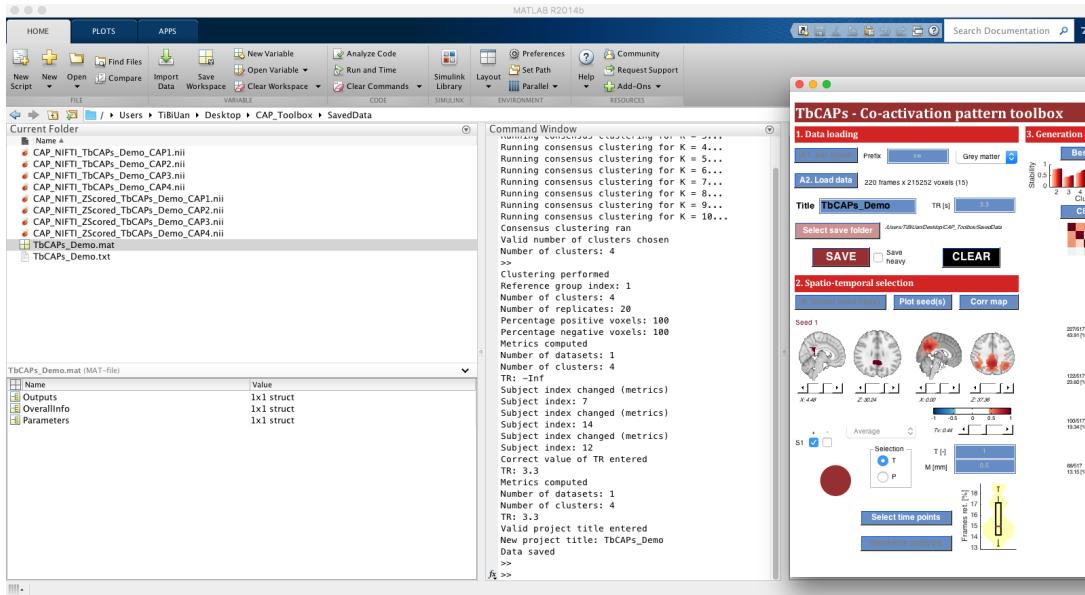


Figure 9: Locating saved data in the dedicated folder.

2.6. Data saving and analysis

At the end of the analysis, we wish to save the results from our computations by clicking on the **SAVE** button. A MATLAB structures can then be found in the chosen save folder, named with the project title, alongside a text file summarising all the inputs to the toolbox (Figure 9). Note that your parameter choices are also displayed in the MATLAB terminal as the analysis goes! In addition to the metrics, two sets of NIFTI volumes containing the CAPs are generated as well: one is the exact equivalent to the toolbox content, where a CAP is simply the average of its constituting frames. The other one is a spatially z-scored variant—by the mode, following Karahanoglu and Van De Ville (2015), so that voxelwise intensity levels can be attributed a probabilistic interpretation.

To look at our computed values, we should first load our saved MATLAB structure by clicking on it; three different structures then become available in the MATLAB workspace: “OverallInfo”, which contains general information on the analyses (such as the data dimensionality); “Parameters”, which summarises all the entered parameters throughout the steps; and “Outputs”, which contains the computed metrics. Note that if the “Save heavy” option is selected in the toolbox interface, a fourth structure will be saved; as can be guessed, it is very heavy, as it contains all the frames that were retained prior to clustering. This is why we leave it up to the user whether to save that information or not, and do not make it a default choice, to avoid slowing down the saving process too much when unnecessary.

In “Outputs.SpatioTemporalSelection” (a subfield of the “Outputs” structure), one finds a matrix tagging the frames that were retained across subjects (“RetainedFramesPerSeed”); the percentage of retained frames for each subject (“PercentageRetainedFrames”); and if computed, a vectorised representation of the average seed correlation map (“AverageCorrelationMap”).

In “Outputs.KMeansClustering”, one finds a measure of dispersion for each CAP (“CoActivationPatternsDis-

Step	Time taken
Mask selection	17 s
Loading of the subjects	9 min 35 s (38.3 s per subject)
Loading of the seed	15 s
Computation of the average seed correlation map	8 s
Selection of time points ($T = 1$, $M = 0.5$ mm)	2 s
Consensus clustering ($K_{\max} = 10$, $P_{CC} = 90\%$, $N = 20$)	1 hour 26 min 13 s (8.62 min per K value)
Clustering into CAPs ($K = 5$, $n_{\text{rep}} = 20$, $P_P = P_N = 100\%$)	11 min 1 s
Computation of CAP metrics	3 s

Table 1: **Time taken by the different processing steps within TbCAPs.** It can be seen that consensus clustering is, by far, the most time consuming step, followed by the extraction of CAPs and the loading of the data.

ersion”, which is the average spatial correlation between the frames assigned to a CAP, and that CAP); the vectorised CAP patterns (“CoActivationPatterns”); their spatially z-scored equivalents (“CoActivationPatternsZScored”); the voxelwise standard deviation for each CAP across assigned frames (“CoActivationPatternsSTD”); and the indices of the CAP to which all retained frames were assigned (“AssignmentsToCAPs”).

In “Outputs.Metrics”, one finds a matrix summarising when CAPs are expressed across subjects (“CAPExpressionIndices”, where -1 entries reflect scrubbed frames, 0 entries stand for non-retained time points, and $K + 1$ entries denote unassigned frames); the occurrence metric values, both raw and normalised so that the total of counts in a subject across CAPs tallies to 100% (“Occurrences”); the number of entries in a given CAP (“NumberEntries”); the average duration for which a CAP is sustained when entered (“AverageExpressionDuration”); all individual expression durations (“AllExpressionDurations”); the number of entries into each CAP from the baseline seed activity level (“CAPEEntriesFromBaseline”); the number of exits from each CAP to that baseline level (“CAPEExitsToBaseline”); measures of CAP resilience (“CAPResilience”), and baseline resilience (“BaselineResilience”); the betweenness centrality metric (“BetweennessCentrality”); in-degree (“CAPInDegree”) and out-degree (“CAPOutDegree”) information; and subject-specific counts of expressed CAPs (“SubjectCounts”).

Please note that for some of these metrics, only CAP values are provided, in which case a dimensionality of K is seen. In other cases, the dimensionality is instead $K + 3$, because we provide information about scrubbed frames (index 1), moments of baseline seed activity (index 2), CAPs (indices 3 to $K + 2$), and unassigned frames (index $K + 3$). Another point worth noting is that if computations were performed for more than one population (see below for details), the results for each of them will lie in a different cell from the MATLAB structure element. Finally, there is an additional output metric named “FractionCAPFramesPerSeedCombination”, which is only relevant in some cases of multi-seed analysis, and specifies how many frames from each CAP were obtained from given combinations of individual seed activity levels.

2.7. Computational time details

To give you some indications on what to expect regarding computational time, in Table 1, we summarise the time taken by each step of CAP analysis using TbCAPs, for the example introduced above. Recall that this particular dataset included 15 subjects, each with 220 functional volumes. Upon masking, 215252 voxels were kept for the analyses. The following values were obtained running MATLAB version 2014b on a MacBook Pro (Retina, 15-inch, Mid 2014), running with a 2.5 GHz Intel Core i7 processor.

3. Additional toolbox functionalities

In what follows, we elaborate on three additional options that may be used, on top of the previous guided example, within TbCAPs.

3.1. Multi-group analysis

In many occasions, we wish to compare different groups of subjects, such as healthy volunteers and a clinically impaired population, in terms of functional brain dynamics. In the above, we showed how the analyses could be performed with a single population of subjects, but up to 4 different populations can successively be loaded through the interface, by simply clicking on the **A2. Load data** button every time, and entering the data of the next population. Importantly, **the content of the toolbox beyond data loading will be cleared every time a new population is loaded**, so we strongly advise you to load all populations in one go at the start of your analyses!

CAPs will always be computed on the first loaded population of subjects: if you wish to derive CAPs from your combined healthy/clinical set of subjects, you should thus select all of them together in one go. If, on the other hand, you are interested in deriving CAPs from only one population (*e.g.*, because you assume that the patterns of activity will strongly differ in diseased subjects), then frames from the other populations should be assigned to the right CAPs. This is done by a matching process: first, each such frame has its spatial correlation with its most similar CAP computed. Second, this value is compared to the distribution of spatial correlations between the CAP in question, and its constituting frame: the assessed frame will only be assigned to the CAP if spatial correlation is sufficiently high compared to this distribution.

The parameter A_P , which should be set in the interface, governs the stringency of this process: if $A_P = 5\%$, for example, a frame will be assigned if its spatial similarity to the CAP exceeds the 5th percentile of the distribution. Thus, selecting a larger A_P value will assign less frames to the CAPs; as a consequence, there will be more occurrences of “Unassigned” frames, which are separately stored in the computed outputs as touched upon above.

Figure 10 shows the final look of the interface when splitting the 15 subjects from the above example into two populations of 8 and 7 subjects, respectively, and selecting $A_P = 10\%$ for assignment. Note that assignment can be rerun as many times as desired with different parameter values by clicking on the **Assign** button.

It can be seen that there are now two sets of violin plots/box plots in the displays—one per population. Furthermore, notice that the values from population 2 are lower: this is partly because, as explained above, some frames were not assigned to any CAP, thus artificially lowering the metric values. In the top representation, many frames from the second group of subjects (second set of rows from the top) are also colour-coded in orange, denoting the “Unassigned” category. Transition probabilities can only be inspected for group 1, but cumulative expression of the CAPs is provided for both populations.

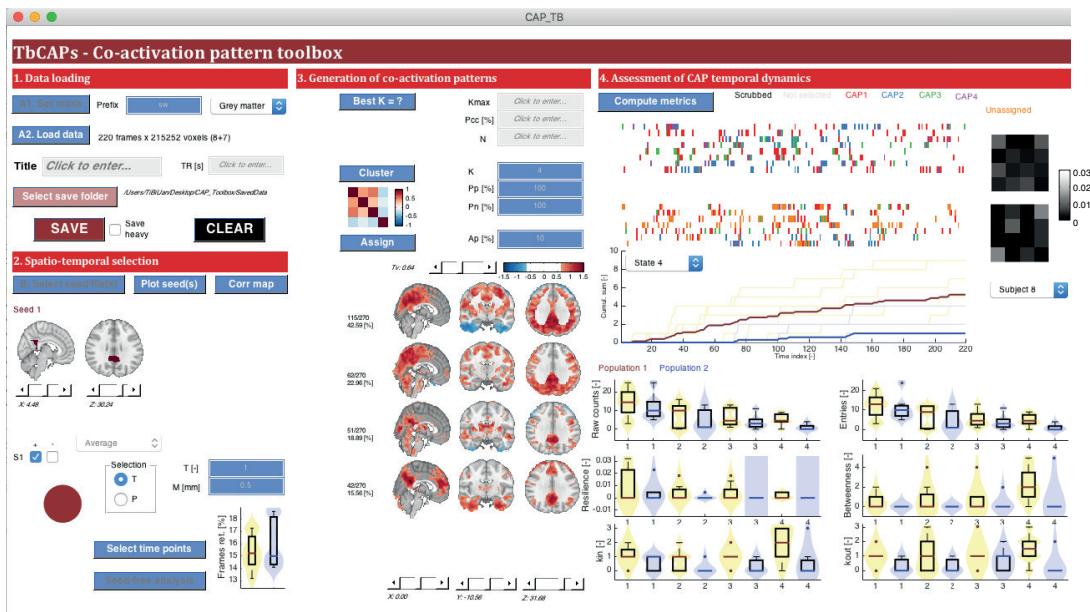


Figure 10: Final state of the interface upon a multi-group analysis.

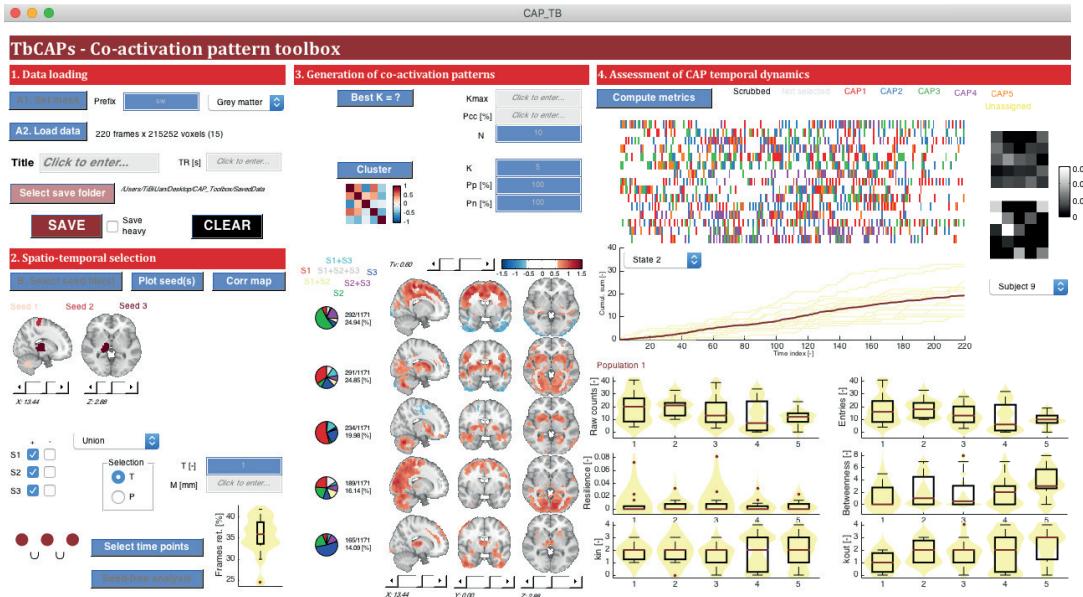


Figure 11: Final state of the interface upon a multi-seed analysis.

3.2. Multi-seed analysis

Another functionality enabled by TbCAPs is to run analyses with up to three seeds at the same time; this can be interesting, for example, if more than one regions of interest are implicated in the network that you wish to study. To run a multi-seed analysis, you simply need to select up to three NIFTI seed files at once upon clicking on the **B. Select seed file(s)** button. Then, the type of event to retain (activation or deactivation) is separately selected for each seed by dedicated check boxes. In addition, you can perform two different types of time point selection: either all seeds should jointly show an extreme signal value (“Intersection” option in the dedicated popup menu), or at least one of the seeds should show such (de)activation (“Union” option). By definition, a larger amount of frames will be retained using the latter.

The only following difference in analytical outcomes is, in the case of the “Union” option, an additional metric: indeed, we can then know, for each CAP, what type of seed combination its constituting frames relate to (*e.g.*, in a two-seed case, perhaps one CAP is mostly generated from frames for which only seed 1 was active, while another is linked to frames showing joint activation between both seeds). This information is depicted as pie charts next to each CAP, and is also saved as the aforementioned “FractionCAPFramesPerSeedCombination” output metric.

Figure 11 displays what the final interface of a “Union” multi-seed analysis looks like, on the same set of 15 subjects as before, when using seeds in the right dental nucleus of the cerebellum (seed 1), the right thalamus (seed 2), and the right precentral gyrus (seed3): notice, for instance, how CAP₅ showcases bilateral thalamic signal, consistent with activation of seed 2, while the associated pie chart indeed confirms that more than half of its constituting frames were selected with only that seed being active.

3.3. Seed-free analysis

Instead of performing a classical seed-based CAP analysis, the toolbox also enables to rather select *all frames* (except those corrupted by excessive motion), without resorting to any seed-based thresholding, following Liu et al. (2013). In this case, the extracted “CAPs” are whole-brain patterns of (de)activation, but the computation process and the output metrics remain similar. To summon this functionality, you should simply click on the **Seed-free analysis** button instead of the **B. Select seed file(s)** one.

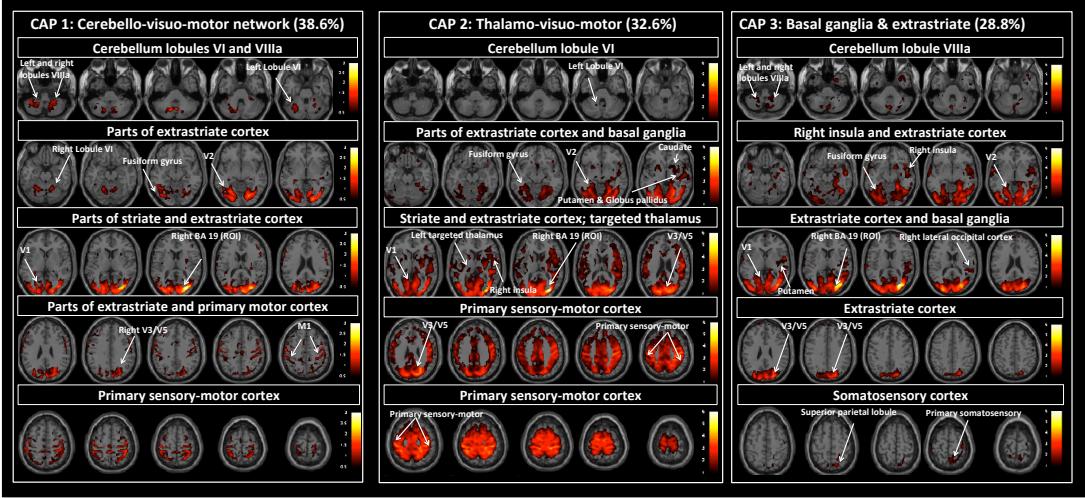


Figure 12: **Extracted co-activation patterns.** Illustration of the 3 CAPs, from 1 to 3 (left to right). Figure reproduced from Tuleasca et al. (2019).

4. Example published application of TbCAPs

In this section, we briefly describe one successful application of TbCAPs to the study of essential tremor (see Tuleasca et al. (2019) for more details). The data in this study was acquired at a TR of 3.3 s, over 10 minutes of resting-state recordings. The final spatial resolution prior to CAP analysis was $2 \times 2 \times 2 \text{ mm}^3$. Three different groups of subjects were considered: (1) healthy controls, (2) patients suffering from essential tremor prior to surgery, and (3) these same patients, one year after having undergone ventral intermediate nucleus (VIM) thalamotomy—the VIM is one of the main brain areas associated to tremor generation. The goals of the study were to determine whether there would be changes in CAP expression following this surgical operation, and if so, how they would relate to the individual extent of tremor symptoms. The right extrastriate cortex, which belongs to the visual circuitry, was used as a seed. This is because recent evidence hinted at an involvement of visual function on tremor extent (Archer et al., 2017).

Time points showing strong seed activation above a threshold $T = 0.5$ were selected, and scrubbing was done at a threshold value $M = 0.5 \text{ mm}$. CAPs were generated from the joint subject population including healthy controls and patients prior to surgery. With consensus clustering, $K = 3$ was selected as an optimum. The three CAPs found in the analysis are displayed in Figure 12. Despite the fact that the seed region belonged to the visual circuitry, many areas associated to essential tremor could be seen in all CAPs. In addition, there were also characteristic patterns to each map (respectively, cerebello-visuo-motor, thalamo-visuo-motor, and basal ganglia/extrastriate cortex).

Following the generation of CAPs, the frames extracted from patients with essential tremor after surgery were assigned to their closest CAP, at $A_P = 5\%$, and CAP metrics were generated for these three subject groups. Only the simplest available metric—the number of occurrences for each CAP—was analysed. Occurrences for all 3 CAPs were altered in patients with essential tremor before surgery, but renormalised back to healthy control levels afterwards, as seen in Figure 13. Further investigations revealed that CAP_2 was expressed less strongly in the more impaired patients pre-surgically, suggesting that it reflects an adaptive trait. In addition,

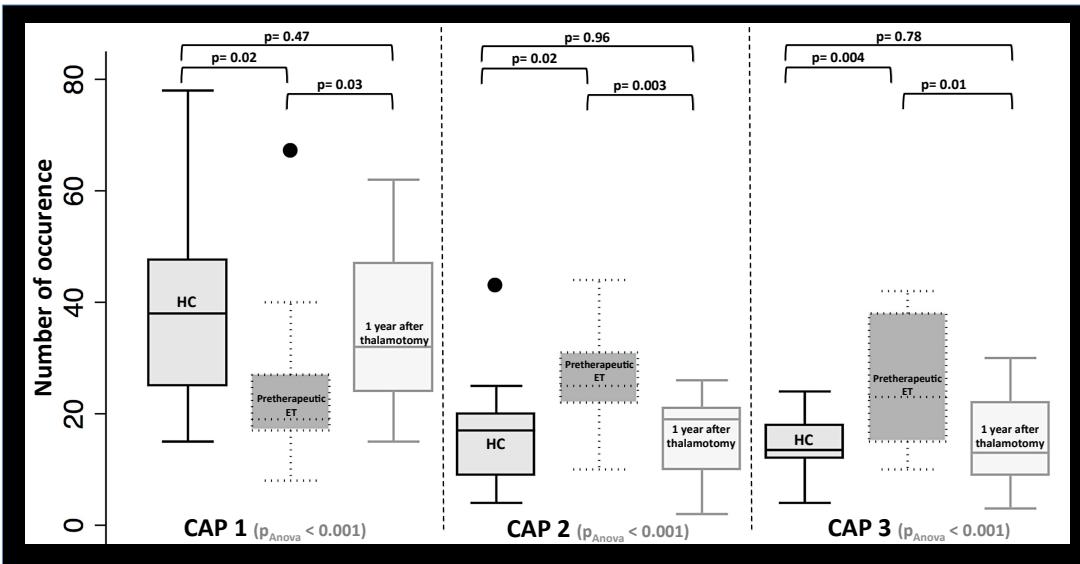


Figure 13: **Renormalisation of occurrences following thalamotomy.** Illustration of the number of occurrences, as box plots, for each CAP across the three investigated groups, with associated p-values. Black circles in CAP₁ and CAP₂ box plots represent outliers. HC: healthy control. ET: essential tremor. Figure reproduced from Tuleasca et al. (2019).

when the expression of CAP₃ was more strongly reduced following surgery, the patients also displayed greater improvements in tremor; thus, this CAP may be part of a compensatory mechanism responding to surgery.

In sum, you can see from this brief description that with TbCAPs, insightful analyses regarding clinical conditions are readily possible. In addition, interesting results do not necessarily require the investigation of overly sophisticated metrics.

5. Methodological subtleties

In this section, we touch upon a few more specific points that we feel are nonetheless important to keep in mind regarding CAP analysis:

A. Choice of the mask to use in the analyses: in TbCAPs, we for now offer only a limited amount of options to mask the input data, two of which are compatible with typical RS analyses: a whole-brain mask, and a grey matter mask. From our past experience, we clearly recommend to make use of the grey matter mask, for several reasons: first, retaining less voxels in the analysis will speed up computations. Second, as we are directly analysing BOLD signals, strong susceptibility to physiological artefacts can be expected, and these are likely to exert the strongest influence on white matter and cerebrospinal fluid brain compartments, which should thus be discarded to avoid as many analytical biases as possible. Even if the data is preprocessed as cleanly as possible, it still makes little sense to add in voxels that will, most likely, only reflect noise. Of course, some users may believe that, in particular, the white matter does contain monitorable signal of neural relevance as well, as has been reported in some occasions—see, *e.g.*, Yarkoni et al. (2009). For such researchers, we also offer the option to analyse the data with a white matter mask.

B. Selecting activation as opposed to deactivation frames: in TbCAPs, the user can decide whether to retain the moments associated to strong seed activation, or deactivation, but cannot select both at the same time. This is because in past work (Liu and Duyn, 2013), some spatial differences have been reported between CAPs generated from one or the other type of frames. In addition, the nature of BOLD deactivation events is unclear: does it reflect lowered local brain activity, or instead, other more sophisticated phenomena such as synchronisation of neural populations at a lower firing frequency, or even the lowering of inhibitory brain activity (which, in practice, makes the BOLD signal go down while electrical activity goes up)? From our past experience, and given the lack of a clear consensus, we would recommend to perform CAP analysis using only the frames featuring seed activation, except in a few specific settings (for example, it may be desirable to select time points when a DMN seed is active, while a task-positive network seed is inactive, as both networks are known to be anti-correlated (Fox et al., 2005) and the extent of this anti-correlation may be clinically insightful—see, for instance, Wotrubá et al. (2014)).

C. Best analytical practices: while the output metrics from CAP analysis are relatively straightforward to analyse, care should be taken not to neglect important covariates of no interest that modulate such values. A first such parameter is the number of extracted time points, which may vary across subjects if the threshold mode (parameter T) is used in selecting time points (*i.e.*, the seed region may become active more frequently in some subjects than others). It should be ensured that, for example, a group difference in CAP occurrences is not caused by this factor alone. Another important concern pertains to multi-group analyses, where the frame assignment process is performed. By design, this process will deem some frames “Unassigned”, and will thus lower CAP occurrences in all groups except the first one. Thus, if a group difference is observed, it should be explicitly verified that it does not come from this artificial effect alone.

6. Other future perspectives of use

Currently, TbCAPs has been designed with the plan of applying it on RS data, in specific experimental settings, as described above. However, we think that there are many other exciting ways to capitalise on the toolbox content, to tackle different aspects of functional brain dynamics. In what follows, we briefly expose a few such ideas of ours, which we hope can serve as a source of inspiration for future users willing to study something new.

A. Impact of acquisition parameters, and preprocessing choices, on the generation of CAPs: co-activation patterns are computed from single fMRI volumes, and the content of such a volume crucially depends on acquisition parameters. In particular, this involves the TR at hand (a shorter TR may imply lower signal-to-noise ratio, but may also enable to resolve faster paced information), and the type of sequence used (for instance, single-band as opposed to multi-band). To date, there are to our knowledge no studies clarifying to what extent such variability may distort co-activation patterns. Another essential aspect that should be further investigated is the impact of preprocessing choices to the generation of CAPs, and their features. It is now well acknowledged that dynamic functional connectivity approaches are particularly sensitive to confounds such as head motion (Laumann et al., 2016), and the impact of physiology on BOLD time courses is also increasingly recognised (Birn, 2012; Liu et al., 2017). Such effects may be expected to be even more potent when it comes to dynamic approaches relying on single frames. Extensive comparative assessments are thus warranted.

B. Application of CAPs to the task-based and real-time settings: while co-activation pattern analysis has so far been largely conducted on resting-state recordings, there is no reason why it could not be applied to task-based studies as well. A first way to do so could be to investigate the resting-state segments that directly follow particular tasks of interest—see, for instance Gaviria et al. (2019) for an analysis of the aftermath of affective and cognitive events. A second equally straightforward possibility lies in the direct analysis of a task, trying to track how CAP expression develops over the course of the paradigm. In addition, there could be an interesting application of the assignment process implemented in TbCAPs in the real-time fMRI case: assuming that a given whole-brain template map is known, one could attempt to locate the time points when brain activity maximally resembles this pattern, and use this information as feedback to the subject being scanned.

7. How to contribute to TbCAPs

While we are very happy to be able to provide a working version of our toolbox, there remains room for future improvements. We will be striving to improve TbCAPs over the following months, and we would be delighted to benefit from the help of motivated users in doing so. Below, we outline the main axes that we envisage when it comes to future adjustments to TbCAPs.

A. Full compatibility with SPM: for the moment, some SPM functions are executed within TbCAPs, but it would be ideal to reach a stage where the toolbox could be directly launched from within SPM, in a fully integrated manner.

B. Compatibility of the data loading with 4D NIFTI files and with IMG/HDR data: sometimes, preprocessed fMRI data takes other formats than a series of 3D NIFTI files, and it would be good to also enable the loading of alternative data formats.

C. Improved mask generation: for the moment, only template masks borrowed from the DPARSF toolbox (Yan and Zang, 2010) are selectable by the user. A more accurate solution, which would also potentially further lower the amount of input voxels to the analysis, could be to compute the mask in a dataset-specific manner: the user would select the runs to jointly analyse, and the associated anatomical information (more precisely, the probabilistic grey matter maps) would be binarised and combined to yield the mask.

D. Loading of intermediate stage data: for the moment, it is not possible to, say, perform the selection of time points (which takes time for large size datasets), save the resulting data, and directly reload that data at a later date without having to run through all the first steps from the toolbox again. Adding a “LOAD” button would enable to improve in that regard.

E. Extended multi-seed analysis: the current multi-seed option could be extended to enable more than 3 separate regions of interest to be jointly analysed. In an ideal world, theoretical concepts would support this extension, and enable to bridge a concrete link to the other extreme that is seed-free analysis.

F. Implementation of alternative data correction methods: as alluded to above, physiological artefacts exert a strong influence on BOLD data, and head motion is not the only confound that should be properly taken care of in the analyses. An interesting direction to follow could be the design of “scrubbing-like” strategies that would operate not from a framewise displacement time course, but instead, at the level of time courses reflective of physiological parameters (*e.g.*, respiration). This should of course be complemented by dedicated analyses looking into the extent to which extreme values of such physiological regressors actually corrupt the BOLD time courses.

G. Adjustable CAP displays: for the moment, only the first five CAPs (with most population-wise occurrences) are displayed in TbCAPs. The interpretation of the data would be facilitated if instead, any of the K computed CAPs could be displayed, as selected through dedicated popup menus.

H. Addition of spatial CAP metrics: in some past works—for instance, ?, instead of temporal metrics reflective of CAP expression, voxelwise spatial differences were contrasted across groups, separately for each CAP, through statistical testing. The addition of such metric computations to the toolbox—which could perhaps be facilitated by the integration of the toolbox within SPM—would broaden the scope of extracted dynamic information by TbCAPs.

I. Extraction of CAP sequences: since different CAPs appear to follow each other in well defined sequences, it could be interesting to try and extract not CAPs, but CAP sequences, starting from a given seed excursion.

The simplest way to do so would be to perform similar computations as for now on a concatenated temporal series of fMRI volumes.

8. FAQ

“MATLAB keeps crashing when I try to process my dataset; I cannot get to the end of these analyses!”

Some of the steps involved in CAP analysis can be computationally very demanding; in particular, this is the case of consensus clustering. In addition, if you work with a large population of subjects, you may reach the memory limits enabled by MATLAB, which is why it crashes. If you are in this situation, one workaround could be, if affordable, to lower the spatial resolution of your data prior to loading it in the toolbox. If you cannot undertake such a drastic adjustment, another option could be to try and run the toolbox interface (or its companion scripts) on a server enabling to work with larger loads of data.

“My data loads properly, but when I try to select time points, I get an error!”

This may indicate that although your data was arranged in the right format to be loaded, it was not preprocessed correctly: for example, perhaps the seed time course that you try to threshold is flat, hence why the time point selection step does not work? This could, for instance, arise if you did not warp your data to MNI space properly.

“I manage to compute CAP metrics, but the results look bizarre, as a given CAP is only expressed at the very start or end of my sequences!”

This is indeed not normal: in theory, you would expect that all CAPs are repeatedly expressed throughout your acquisition. A common source for such an issue is the lack of a detrending step: spurious linear increases or decreases in the BOLD time courses may then, by themselves, drive the generation of some of the CAPs (for example, a CAP may reflect the presence of very low signal overall at the start of acquisition, hence why it would be located there in the metrics plot).

“I find no significant group difference despite all the time I invested in these stupid analyses!”

First, take a deep breath: research is hard, and often cruel, but what matters above all is that you always try to do your very best. After a resourceful break, you should first double check every step of your analyses, to make sure that you performed them properly. If it indeed holds that there is no difference between your groups, then so be it: however, always remember that a negative research finding is actually just as insightful as the opposite, and that those who disagree are probably obsessed with their publication count. You are worth much better than that!

References

- Archer, D. B., Coombes, S. A., Chu, W. T., Chung, J. W., Burciu, R. G., Okun, M. S., Wagle Shukla, A., Vaillancourt, D. E., 2017. A widespread visually-sensitive functional network relates to symptoms in essential tremor. *Brain* 141 (2), 472–485.
- Birn, R. M., 2012. The role of physiological noise in resting-state functional connectivity. *Neuroimage* 62 (2), 864–870.
- Bolton, T. A., Tuleasca, C., Rey, G., Wotruska, D., Gaviria, J., Dhanis, H., Blondiaux, E., Gauthier, B., Smigelski, L., Van De Ville, D., 2019. Tbcaps: A toolbox for co-activation pattern analysis. arXiv preprint arXiv:1910.06113.
- Fox, M. D., Snyder, A. Z., Vincent, J. L., Corbetta, M., Van Essen, D. C., Raichle, M. E., 2005. The human brain is intrinsically organized into dynamic, anticorrelated functional networks. *Proceedings of the National Academy of Sciences* 102 (27), 9673–9678.
- Gaviria, J., Rey, G., Bolton, T., Delgado, J., Van de Ville, D., Vuilleumier, P., 2019. Brain functional connectivity dynamics in the aftermaths of affective and cognitive events. *bioRxiv*, 685396.
- Gutierrez-Barragan, D., Basson, M. A., Panzeri, S., Gozzi, A., 2019. Infraslow state fluctuations govern spontaneous fmri network dynamics. *Current Biology* 29 (14), 2295–2306.
- Hutchison, R. M., Womelsdorf, T., Allen, E. A., Bandettini, P. A., Calhoun, V. D., Corbetta, M., Della Penna, S., Duyn, J. H., Glover, G. H., Gonzalez-Castillo, J., Handwerker, D. A., Keilholz, S., Kiviniemi, V., Leopold, D. A., de Pasquale, F., Sporns, O., Walter, M., Chang, C., 2013. Dynamic functional connectivity: Promise, issues, and interpretations. *NeuroImage* 80, 360–378.
- Jenkinson, M., Beckmann, C. F., Behrens, T. E. J., Woolrich, M. W., Smith, S. M., 2012. Fsl. *Neuroimage* 62 (2), 782–790.
- Karahanoğlu, F. I., Van De Ville, D., 2015. Transient brain activity disentangles fMRI resting-state dynamics in terms of spatially and temporally overlapping networks. *Nature Communications* 6, 7751.
- Kodinariya, T. M., Makwana, P. R., 2013. Review on determining number of cluster in k-means clustering. *International Journal* 1 (6), 90–95.
- Laumann, T. O., Snyder, A. Z., Mitra, A., Gordon, E. M., Gratton, C., Adeyemo, B., Gilmore, A. W., Nelson, S. M., Berg, J. J., Greene, D. J., et al., 2016. On the stability of bold fmri correlations. *Cerebral Cortex* 27 (10), 4719–4732.
- Liu, T. T., Nalci, A., Falahpour, M., 2017. The global signal in fmri: Nuisance or information? *Neuroimage* 150, 213–229.
- Liu, X., Chang, C., Duyn, J. H., 2013. Decomposition of spontaneous brain activity into distinct fMRI co-activation patterns. *Frontiers in Systems Neuroscience* 7, 1–11.
- Liu, X., Duyn, J. H., 2013. Time-varying functional network information extracted from brief instances of spontaneous brain activity. *Proceedings of the National Academy of Sciences* 110 (11), 4392–4397.

- Lurie, D., Kessler, D., Bassett, D., Betzel, R. F., Breakspear, M., Keilholz, S., Kucyi, A., Liégeois, R., Lindquist, M. A., McIntosh, A. R., 2018. On the nature of resting fmri and time-varying functional connectivity. *PsyArXiv*.
- Monti, S., Tamayo, P., Mesirov, J., Golub, T., 2003. Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data. *Machine Learning* 52 (1), 91–118.
- Power, J. D., Barnes, K. A., Snyder, A. Z., Schlaggar, B. L., Petersen, S. E., 2012. Spurious but systematic correlations in functional connectivity MRI networks arise from subject motion. *Neuroimage* 59 (3), 2142–2154.
- Preti, M. G., Bolton, T. A. W., Van De Ville, D., 2017. The dynamic functional connectome: State-of-the-art and perspectives. *Neuroimage* 160, 41–54.
- Rousseeuw, P. J., 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics* 20, 53–65.
- Senbabaoglu, Y., Michailidis, G., Li, J. Z., 2014. Critical limitations of consensus clustering in class discovery. *Scientific Reports* 4, 6207.
- Shukla, S., Naganna, S., 2014. A review on k-means data clustering approach. *International Journal of Information and Computation Technology* 4 (17), 1847–1860.
- Tagliazucchi, E., Balenzuela, P., Fraiman, D., Chialvo, D. R., 2012. Criticality in large-scale brain fMRI dynamics unveiled by a novel point process analysis. *Frontiers in Physiology* 3, 1–12.
- Tuleasca, C., Bolton, T. A. W., Régis, J., Najdenovska, E., Witjas, T., Girard, N., Delaire, F., Vincent, M., Faouzi, M., Thiran, J., et al., 2019. Normalization of aberrant pretherapeutic dynamic functional connectivity of extrastriate visual system in patients who underwent thalamotomy with stereotactic radiosurgery for essential tremor: a resting-state functional mri study. *Journal of Neurosurgery* 1, 1–10.
- Tzourio-Mazoyer, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., Mazoyer, B., Loliot, M., 2002. Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. *NeuroImage* 15, 273–289.
- Wotruba, D., Michels, L., Buechler, R., et al., 2014. Aberrant coupling within and across the default mode, task-positive, and salience network in subjects at risk for psychosis. *Schizophrenia Bulletin* 40 (5), 1095–1104.
- Yan, C., Zang, Y., 2010. DPARSF: a MATLAB toolbox for "pipeline" data analysis of resting-state fMRI. *Frontiers in Systems Neuroscience* 4, 13.
- Yarkoni, T., Barch, D. M., Gray, J. R., Conturo, T. E., Braver, T. S., 2009. Bold correlates of trial-by-trial reaction time variability in gray and white matter: a multi-study fmri analysis. *PLOS ONE* 4 (1), e4257.
- Zöller, D. M., Bolton, T. A. W., Karahanoglu, F. I., Eliez, S., Schaer, M., Van De Ville, D., 2018. Robust recovery of temporal overlap between network activity using transient-informed spatio-temporal regression. *IEEE Transactions on Medical Imaging* 38 (1), 291–302.