Supplementary Material

Real-time estimation of dynamic functional connectivity networks

Ricardo Pio Monti, Romy Lorenz, Rodrigo M. Braga, Christoforos Anagnostopoulos, Robert Leech & Giovanni Montana

Supplementary Material

A. Full derivation of Adaptive filtering derivative

The results shown in this section are taken from Anagnostopoulos et al. (2012).

The log-likelihood for unseen observation, X_{t+1} is given by:

$$\mathcal{L}_{t+1} = \mathcal{L}(X_t;\ \bar{x}_t, S_t) = -\frac{1}{2}\log\det S_t - \frac{1}{2}\,(X_{t+1} - \bar{x}_t)^T S_t^{-1}(X_{t+1} - \bar{x}_t).$$

The approach taken here is to approximate the derivative of $\mathcal{L}_{t+1}$ with respect to adaptive forgetting factor r_t by calculating the exact derivative of $\mathcal{L}_{t+1}$ with respect to a fixed forgetting factor *r*. Then under the assumption that changes in r_t occur sufficiently slowly, this will serve as a good approximation to the derivative of $\mathcal{L}_{t+1}$ with respect to r_t.

We begin by noting the following results from Petersen and Pedersen (2008):

$$\frac{\partial \log\det S_t}{\partial r} = \text{trace}\,(S_t^{-1}S_t')$$

$$\frac{\partial\, S_t^{-1}}{\partial r} = S_t^{-1}S_t'\,S_t^{-1}$$

Moreover, we note that we do not need to explicitly invert S_t. By noting that S_t is a rank one update of S_{t-1} we are able to directly obtain S_t^{-1} using the Sherman-Woodbury formula.

Further, from equations (2), (4), (5) and (10) we can see that:

$$\begin{aligned}
\overline{x}'_t &= \left(1 - \frac{1}{\omega_t}\right)\bar{x}'_{t-1} + \frac{\omega_t'}{\omega_t^2}(X_t - \bar{x}_{t-1}) \\
\omega_t' &= r_{t-1}\,\omega_{t-1}' + \omega_t \\
\prod{}'_t &= \left(1 - \frac{1}{\omega_t}\right)\prod{}'_{t-1} + \frac{\omega_t'}{\omega_t^2}(X_tX_t^T - \Pi_{t-1}) \\
S_t' &= \prod{}'_t - \bar{x}'_t\,\bar{x}_t^T - \bar{x}_t\,(\bar{x}_t')^T
\end{aligned}$$

where once again we have used the notation *A*' to denote the derivative of a vector or matrix A with respect to *r*. Using the results from the above equations we can directly differentiate the $\mathcal{L}_{t+1}$ to obtain equation (7).

B. Optimization algorithm for rt-SINGLE algorithm

In this section we provide further details regarding the optimization of the rt-SINGLE objective funciton. Equations (12)-(13) clearly expose the separable nature of the objective, which can be expressed as the sum of two sub-functions. It is precisely this property which is exploited in the original SINGLE algorithm by employing an Alternating Directions Method of Multipliers (ADMM) algorithm (Boyd et al., 2010). The ADMM is a form of augmented Lagrangian algorithm that is particularly well suited to addressing this class of separable and highly structured minimization problems. Formally, such an algorithm proceeds by iteratively minimizing each of the sub-functions together with an additional Lagrangian penalty term. As we demonstrate below, each of these minimization problems will either have a closed form solution or can be efficiently solved.

As in the SINGLE algorithm, we proceed by introducing an auxiliary variable $Z \in \mathbb{R}^{p\times p}$. Here Z corresponds directly to K and we require $Z = K$ for convergence. Minimizing equation (11) can subsequently be cast as the following constrained optimization problem:

$$minimize_{K,Z}\left\{-\log\det K + \text{trace}(S_tK) + \lambda_1\left|\left|Z\right|\right|_1 + \lambda_2\left|\left|Z - K_{t-1}\right|\right|_1\right\}\ \text{subject to K=Z} \qquad (14)$$

We note that *K* is now only involved in the likelihood component while *Z* is involved exclusively in the penalty component. Thus, by introducing *Z* we have decoupled the initial objective function — allowing us to take advantage of the individual structure associated with each term.

We formulate the augmented Lagrangian corresponding to equation (14), which is defined as:

$$\mathcal{L}_\gamma(K,Z,U) = -\log\det K + \text{trace}\,(S_tK) + \lambda_1\left|\left|Z\right|\right|_1 + \lambda_2\left|\left|Z - K_{t-1}\right|\right|_1 + \frac{1}{2}\left(\left|\left|K - Z + U\right|\right|_2^2 - \left|\left|U\right|\right|_2^2\right) \qquad (15)$$

where $U \in \mathbb{R}^{p\times p}$ is the (scaled) Lagrange multiplier. Equation (15) corresponds to the Lagrangian together with an additional quadratic penalty term which serves to both increase the robustness of the proposed method (Bertsekas, 1982) as well as greatly simplify the resulting computations, as we describe below. The proposed estimation algorithm works by iteratively minimizing equation (15) with respect to *K* and *Z*. In this manner, the proposed algorithm is able to decouple the objective function of equation (11), leading to simple sub-problems.

B.1 Burn-in period

It is common for real-time algorithms to incorporate a brief burn-in phase a when they are initialized. This involves collecting the first N_{burnIn} observations and using these to collectively obtain the first estimate. Many times such an approach is motivated by the need to ensure sample statistics are well-defined, however due to the presence of regularization the proposed method does not require a burn-in *per se*. That said the use of a burn-in phase can improve initial network estimates and thereby result in improved network estimation initially. As a result, the first N_{burnIn} observations are collected and used to estimate the corresponding precision matrices by directly applying the offline SINGLE algorithm. This involves solving equation (**Error! Reference source not found.**0). From then onward, new estimates of the precision matrix are obtained as described previously.

C. Additional information regarding simulation study setting

C.1 Simulation settings

In each simulation we produce simulated time series data giving rise to a number of connectivity patterns and properties which reflect those reported in real fMRI data. The objective is then to measure whether our proposed algorithm is able recover the underlying patterns in real-time. We are primarily interested in studying the performance of the proposed methods in two ways; first we wish to study the quality of the estimated covariance matrices over time. That is to say, we study how accurately our sample covariances represent the true underlying covariance structure. Second, we are also interested in the correct estimation of the presence or absence of edges.

There are two main properties of fMRI data which we wish to recreate in the simulation study. The first is the high autocorrelation which is typically present in fMRI data (Poldrack et al., 2011). The second and main property we wish to recreate is the structure of the connectivity networks themselves. It is widely reported that brain networks have a small-world topology as well as highly connected hub nodes (Bullmore and Sporns, 2009) and we therefore look to enforce these properties in our simulated data.

C.1.1. Autocorrelation

Vector Autoregressive (VAR) processes are well suited to the task of producing autocorrelated multivariate time series as they are capable of encoding autocorrelations within components as well as cross correlations across components (Cribben et al., 2012). The focus of these simulations is to study the performance of the proposed method in the presence of non-stationary data. As a result the simulated datasets are only locally stationary. This is achieved by concatenating multiple VAR process which are simulated independently — this results in abrupt changes which are representative of the typical block structure of task-based fMRI experiments. Formally, data is generated as follows: for each block the covariance structure graph is randomly sampled. This is subsequently employed to simulate data from a VAR model which enforces cross-correlations of the simulated graph. This results in highly autocorrelated data which exhibits the conditional independence structure of the simulated graph.

C.1.2 Grap theoretical properties

Moreover, when simulating connectivity structures we study the performance of the proposed algorithm using two types of random graphs; scale-free random graphs obtained by using the preferential attachment model of Barabási and Albert (1999) and small-world random graphs obtained using the Watts and Strogatz (1998) model. The use of each of these types of networks is motivated by the fact that they are each known to each resemble different aspects of fMRI networks.

Throughout each of the simulations, first the network architecture was simulated using either of the aforementioned methods. Then edge strength was uniformly sampled from $\left[-\frac{1}{2}, -\frac{1}{4}\right] \cup \left[\frac{1}{4}, \frac{1}{2}\right]$ This introduced further variability into the simulated networks, increasing the difficulty of each task.

C.1.3 Parameter tuning

The parameters for the offline SINGLE algorithm where determined as described in Monti et al. (2014). That is, the choice of kernel width was obtained by maximizing leave-one-out log-likelihood while the choice of regularization parameters where chosen by minimizing AIC. In the case of the real-time algorithms the parameters where chosen as follows. The sliding window length and fixed forgetting factor were selected in order to have approximately equal sample size (provided in equation (16)). As such, a window length of 40 was employed and the fixed forgetting factor was set to be r=0.975. While in the case of adaptive forgetting η=0.005 was chosen. All real-time algorithms employed a burn-in of 15 observations. Regularization parameters where chosen to minimize AIC over the burn-in period.

C.2 Performance measures

As alluded to previously, we wish to evaluate the performance of the proposed method in two distinct ways. First, we wish to study the reliability with which we can track changes in covariance structure using either a fixed forgetting factor or an adaptive forgetting factor. In order to quantify the difference between the true covariance structure and our estimated covariance we consider the distance defined by the trace inner product:

$$d(\sum, S) = \text{trace}(\sum^{-1} S)$$

It follows that if the estimated sample covariance, S, is a good estimate of the true covariance, $\sum$, we will have that $d(\sum, S) \approx p$. However, if S is a poor estimate, the distance d will be large. Moreover, since both $\sum$ and S are positive definite we have that $d(\sum, S)$ will always be positive.

Second, we wish to consider the estimated functional connectivity networks at each point in time. In this application we are particularly interested in correctly identifying the non-zero entries in estimated precision matrices, $\widehat{K}_t$, at each i=1,…,T. An edge is assumed to be present between the jth and kth nodes if $\left(\widehat{K_t}\right)_{j,k} \neq 0$. At the ith observation we define the set of all reported edges as $D_i = \{(j,k) : \left(\widehat{K}_i\right)_{j,k} \neq 0\}$. We define the corresponding set of true edges as $T_i = \{(j,k) : (K_i)_{j,k} \neq 0\}$. where we write K_i to denote the true precision matrix at the ith observation. Given D_i and T_i we consider a number of performance measures at each observation.

1. (1) We measure the precision, P_i. This measures the percentage of reported edges which are actually present (i.e., true edges). Formally, the precision is given by: $P_i = \frac{|D_i \cap T_i|}{|D_i|}$.
2. we also calculate the recall, R_i, formally defined as:

$$R_i = \frac{|D_i \cap T_i|}{|T_i|}.$$

   This measures the percentage of true edges which were reported by each algorithm. Ideally we would like to have both precision and recall as close to one as possible.

3. Finally, the F_i *score*, defined as:

$$F_i = 2\,\frac{P_i R_i}{P_i + R_i},$$

   summarizes both the precision and recall by taking their harmonic mean. It follows that F_i will lie on the interval [0,1] with $F_i = 1$ indicating perfect performance.

Supplementary Tables

Table 1: Eleven regions of interest and corresponing MNI coordinates (center of gravity) used in the study of HCP data described in Section 4.1

Name of ROI	Right hemisphere			Left hemisphere		
Lateral Occipital	31	-84	1	-29	-87	1
Inferior Parietal	43	-62	30	-39	-68	30
Superior Parietal	22	-62	48	-21	-64	47
Precuneus	11	-56	37	-10	-57	37
Fusiform	34	-39	-20	-34	-43	-19
Lingual	15	-66	-3	-14	-67	-3
Inferior Temporal	49	-26	-25	-49	-31	-23
Middle Temporal	57	-22	-14	-56	-27	-12
Precentral	39	-8	43	-38	-9	43
Postcentral	42	-21	44	-42	-23	44
Paracentral	9	-26	58	-8	-28	59

Table 2: Summary graph statistics (sd) for networks estimated using the offline and real-time SINGLE algorithms respectively. Graph statistics are provided for task positive and task negative networks (correspond to red and blue edges in Figure [6] respectively) in order to allow for a detailed study of graph properties across both algorithms.

	Offline SINGLE		Real-time SINGLE	
Graph Statistic	Task positive	Task negative	Task positive	Task negative
Degree centrality	0.29 (0.10)	0.09 (0.05)	0.27 (0.11)	0.07 (0.04)
Betweenness centrality	0.07 (0.02)	0.01 (0.01)	0.1 (0.03)	0.02 (0.01)
Transitivity	0.24	0.06	0.22	0.06

Table 3: Summary graph statistics (sd) for networks estimated using the real-time SINGLE algorithm across the cohort of subjects. Graph statistics are provided for task positive and task negative networks (correspond to red and blue edges in Figure [7] respectively) in order to allow for a detailed study of the robust nature of graph statistics across all subjects.

	Estimated value across subjects	
Graph Statistic	Task positive	Task negative
Degree centrality	0.31 (0.05)	0.11 (0.06)
Betweenness centrality	0.06 (0.01)	0.01 (0.01)
Transitivity	0.27 (0.06)	0.08 (0.04)

Table 4: Fourteen regions of interest and corresponing MNI coordinates used in the real-time fMRI study described in Section 4.2

Name of ROI	MNI coordinates		
Right middle frontal gyrus (RMFG)	33	22	37
Right frontal eye fields (RFEF)	24	13	52
Left inferior frontal gyrus (LIFG)	-45	32	15
Right inferior frontal gyrus (RIFG)	49	20	19
Right superior parietal lobe (RSPL)	30	-65	49
Left superior parietal lobe (LSPL)	-23	-67	46
Left occipital fusiform	-10	-83	-23
Right occipital fusiform	29	-91	-17
Left frontal pole	-41	53	-1
Right inferior temporal gyrus (RITG)	58	-52	-18
Right middle temporal gyrus	62	-37	-8
Superior parietal lobe	23	-65	48
Frontal eye fields	32	-10	48
Middle frontal gyrus	46	6	42

Table 5: Summary graph statistics (sd) for networks estimated using the real-time SINGLE algorithm across the cohort of subjects. Graph statistics are provided for networks associated with attentive visual search and passive viewing respectively (correspond to red and blue edges in Figure [8] respectively).

	Estimated value across subjects	
Graph Statistic	Attentive visual search	Passive viewing
Degree centrality	0.39 (0.11)	0.18 (0.06)
Betweenness centrality	0.07 (0.02)	0.02 (0.01)
Transitivity	0.5 (0.08)	0.06 (0.03)

Supplementary Figures

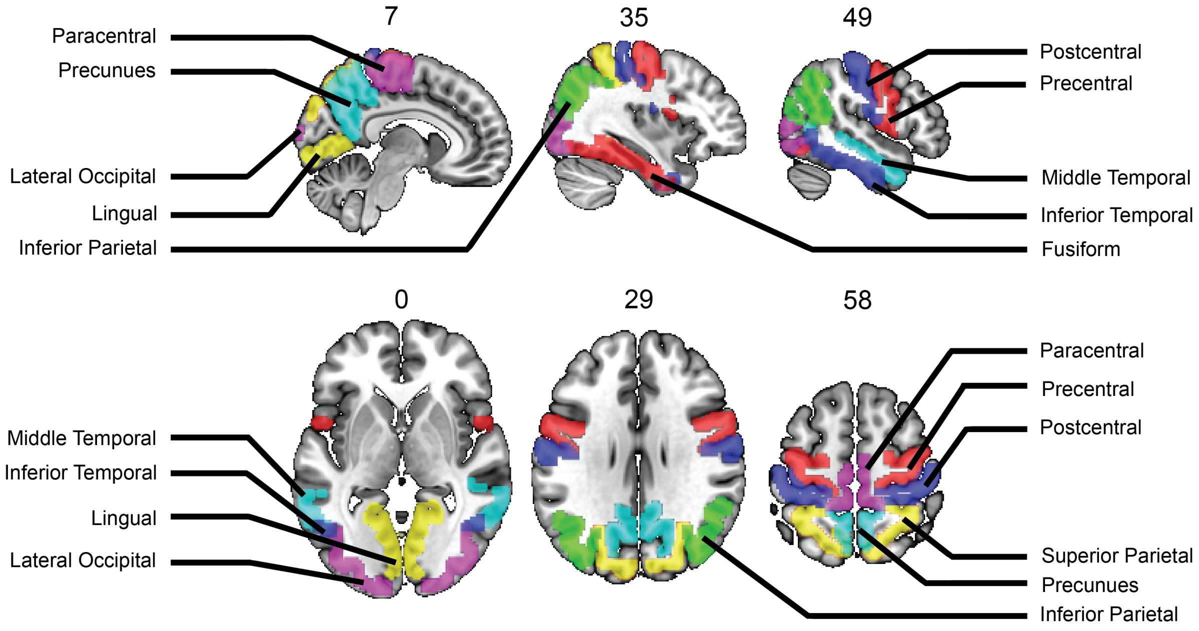

Figure 1: Description of the eleven bilateral ROIs used in the study of HCP data described in Section 4.1