A probabilistic approach to full-brain functional connectivity analyses


*Princeton University, Princeton NJ, †New York University, ‡University of Guelph, and §Columbia University, New York NY

Submitted to Proceedings of the National Academy of Sciences of the United States of America

Recent work suggests that our brains’ sub-structures change how they communicate with one another depending on the particular functions or computations that our ongoing cognitive processes demand (for review see [1]). The standard approach to estimating these so called functional connectivity patterns in functional magnetic resonance imaging (fMRI) data is to compute the correlation between the time series of (pairs of) voxel activations during a particular experimental condition. However, this voxel-based approach carries a substantial computational burden (of computing time and memory), which has led most researchers to focus their connectivity analyses on a small number of pre-selected regions of interest (ROIs). Here we present a technique, termed Hierarchical Topographic Factor Analysis (HTFA), for efficiently discovering full-brain networks (without pre-selecting ROIs) in large multi-subject neuroimaging datasets. HTFA approximates each subject’s full-brain functional connectivity network through a smaller number of network nodes. The number of nodes, along with their locations, sizes, and activations (over time) are determined in an unsupervised manner from the dataset. Because the number of nodes is typically substantially smaller than the number of voxels in an fMRI dataset, HTFA can be orders of magnitude more efficient than voxel-based functional connectivity approaches. Among other benefits, this enables researchers to apply polynomial time algorithms (which includes many pattern classification algorithms) to full-brain functional connectivity networks without paying the typical huge increase in computational time and memory that voxel-based methods demand. We show that HTFA recovers the connectivity patterns underlying a synthetic dataset, and provide a case study illustrating how HTFA may be used to discover full-brain connectivity patterns in real fMRI data.

Abbreviations: HTFA, hierarchical factor analysis; PCA, principal components analysis; ICA, independent components analysis; fMRI, functional magnetic resonance imaging; ROI, region of interest; RBF, Gaussian radial basis function; RAM, random access memory

The most common approaches for analyzing functional Magnetic Resonance Imaging (fMRI) data involve relating, in individual images, the activity of individual voxels or multi-voxel spatial patterns of brain activity to the subject’s cognitive state [2, 3, 4, 5]. In contrast, functional connectivity analyses estimate connections between brain regions by correlating the time series of activations across images of pairs of voxels [6]. This cutting-edge approach has already led to important new insights into how the brain’s connections change during different experimental conditions [1].

Because the full-brain connectivity matrix grows with the square of the number of voxels, both filling in its entries and storing it in memory can become intractable for fMRI images with tens of thousands of voxels. For example, the connectivity matrix for a 50000 voxel image occupies approximately 5 GB of memory (assuming single precision floating point entries). Storing many such matrices in memory (e.g. to compare different subjects and/or experimental conditions) can be impractical on modern hardware. Further, many of the algorithms used to relate multivariate patterns of voxel activations in individual images to cognitive representations or experimental conditions (e.g. [7, 5]) use polynomial time and memory (with respect to the number of features), making it impractical to use the same techniques to examine connectivity data (although see [8] for an alternative approach).

Hierarchical Topographic Factor Analysis (HTFA) provides an alternative means of representing the brain’s connectivity patterns that scales well to large datasets. HTFA casts each subject’s brain images as linear combinations of latent factors [Gaussian radial basis functions (RBFs)]. Each RBF can be interpreted as a node in a simplified representation of the brain’s networks. (The number of factors, K, is determined from the data.) Applying HTFA to an fMRI dataset reveals the locations and sizes of these factors (i.e. the centers and widths of their RBFs), as well as the per-image factor weights. If a given subject has contributed N images to the dataset, then the subject’s N by K factor weights matrix may be viewed as a low-dimensional embedding of their original data. Further, the pairwise correlations between columns of this weight matrix reflect the signs and strengths of the node-to-node connections (just as the pairwise correlations between voxel time series reflect the corresponding connectivity in voxel-based functional connectivity analyses).

We designed HTFA to naturally account for data from multiple subjects. The model comprises a global template, which describes how data look in general (e.g. for the “typical” subject), as well as subject-specific templates which describe how each individual subject differs from that global template (Fig. 1). This hierarchical design constrains the RBF centers and widths to be similar across subjects (without forcing them to be the same). Further, since every subject-specific template has an instantiation of every RBF in the global template, the model provides a natural means of performing across-subject hypothesis testing on functional connectivity patterns. In the next section we provide an overview and formal description of the HTFA model, and in Materials and methods we describe how one can efficiently fit the model to a multi-subject fMRI dataset. To demonstrate our approach, we apply HTFA to a spatial attention fMRI experiment that had subjects selectively attend (in each trial) to images of either faces or scenes, either on the left or the right of a computer screen. As an example, we examine how full-brain connectivity changed between the different experimental conditions:

Reserved for Publication Footnotes
face/left, face/right, scene/ left, scene/right. We first generate a synthetic dataset based on this experimental paradigm (whose underlying connectivity patterns, node locations and sizes, etc., are known) and show that HTFA recovers these known patterns. We then apply HTFA to a real multi-subject fMRI dataset and identify face-selective, scene-selective, left-selective, and right-selective brain networks that modulate their connectivity to each other according to the instructions in the different conditions.

Model description

Overview. HTFA is a member of a family of models, called factor analysis models, that includes Topographic Factor Analysis (TFA) [9], Topographic Latent Source Analysis (TLSA) [10], Principal Components Analysis (PCA) [11], Exploratory Factor Analysis (EFA) [12], and Independent Components Analysis (ICA) [13, 14], among others. If we have exploratory Factor Analysis (EFA) [12], and Independent Components Analysis (ICA) [13, 14], among others. If we have organized our collection of images (from a single subject) into an N by V matrix Y (where N is the number of images and V is the number of voxels), then factor analysis models decompose Y as follows:

\[ Y \approx WF, \]

where W is an N by K weight matrix (which describes how each of K factors are activated in each image), and F is a K by V matrix of factor images (which describes how each factor maps onto the brain). Note that, in the typical setting, this decomposition is underspecified—in other words, typically there are infinitely many solutions for W and F that approximate the data equally well. What differentiates factor analysis models is the particular constraints they place on what form W and/or F should take (i.e. by changing the function being optimized in order to settle on a specific choice of W and F). We may then use W as a low-dimensional embedding of the original data (e.g. to facilitate interpretability or improve computational tractability), or we may choose to examine the factor images in F to gain insights into the dataset.

In standard approaches such as PCA and ICA, the entries of W and F are Real numbers. In PCA, each row of F is an eigenvector of the data covariance matrix, and W is chosen to minimize the reconstruction error (i.e. to make WF as close as possible to Y in terms of mean squared error). In ICA, the goal is to minimize the statistical dependence between the rows of F while also adjusting W to minimize the reconstruction error. In this way, the factor images (the rows of F) obtained using PCA and ICA are unstructured images (i.e. activation patterns) of the same complexity as observations in the original dataset: each factor is parameterized by V numbers (1 parameter per voxel).

In TFA (and TLSA, of which TFA is a special case), each row of F is parameterized by the center parameter, µ, and the width parameter, λ, of an RBF. If an RBF has center µ and (log) width λ, then its activation RBF(\(r|\mu,\lambda\)) at location r is given by:

\[ \text{RBF}(r|\mu,\lambda) = \exp \left\{ -\frac{|r-\mu|^2}{\exp(\lambda)} \right\}. \]

The factor images are filled in by evaluating each RBF, defined by the corresponding parameters for each factor, at the location of each voxel. In contrast to the factors obtained using PCA or ICA, TFA’s more constrained factors are easier to interpret and identify face-selective, scene-selective, left-selective, and right-selective brain networks that modulate their connectivity to each other according to the instructions in the different conditions.

HTFA works similarly to TFA, but places an additional constraint over the factors to bias all of the subjects to exhibit similar factors. In this way, whereas TFA attempts to find the factors that best explain an individual subject’s data, HTFA also attempts to find the factors that are common across a group of subjects (Fig. 1). This is an important advance, because it allows the model to jointly consider data from multiple subjects (thereby allowing for across-subject analyses, etc.).

The model handles multi-subject data by defining a global template, which describes in general where each RBF is placed, how wide it is, and how active its factor tends to be. In addition to estimating how factors look and behave in general (across subjects), HTFA also estimates each individual’s subject-specific template, which describes each subject’s particular instantiations of each RBF (i.e. that subject’s RBF locations and widths) and the factor activations (i.e. the activations of each of that subject’s factors in each of that subject’s images). The factor activations, in turn, are used to approximate the full-brain functional connectivity patterns.

Formal definition and notation. We formulate HTFA as a probabilistic model, which can be represented in graphical model notation. In the graphical model (Fig. 2), variables associated with the subject-specific template are found in the yellow plate. These include the subject-specific RBF centers \(\{\mu_k,1:s\}\), RBF widths \(\{\lambda_k,1:s\}\), and per-image factor weights \(\{w_{n1:n1,1:s}\}\), as well as the observed images \(\{y_{1:n1,1:s}\}\). Variables associated with the global template are found outside of the yellow plate; these include the global RBF centers \(\{\mu_k\}\) and the global RBF widths \(\{\lambda_k\}\). The subject-specific templates are conditioned on the global template, thereby associating data from different subjects. (This interaction between the subject-specific and global templates occurs where the yellow and blue plates overlap.)

The structure of the graphical model specifies the conditional dependencies in HTFA:

\[ p(Y_{1:n1}\Omega) = \psi^\gamma, \]

where \(Y_s\) is the \(N_s\) by \(V_s\) matrix of images from subject s, \(\Omega\) is the set of hidden variables in the model

\[ \Omega = \{w_{1:n1,1:k}\}, \mu_{1:k}, \lambda_{1:k}\}, \]

where both the number of images from subject s, \(N_s\), and the number of voxels from subject s, \(V_s\), may vary across subjects.
and where the probability of the global template $\gamma$ is given by
\[ \gamma = \prod_{k=1}^{K} p(\hat{\mu}_k)p(\hat{\lambda}_k). \]  
\[ \text{(Note that the hyperparameters have been omitted to simplify the notation.)} \]

Another useful way to describe HTFA is through the generative process it implies. HTFA’s generative process is an algorithm that, when run, generates a single sample from HTFA’s joint distribution (Eqn. 3), yielding one value for each hidden variable and a multi-subject fMRI dataset. HTFA’s generative process is detailed in Algorithm 1. The generative process starts by drawing a set of global RBF parameters, and from there draws subject-specific RBF parameters and factor weights, and from there draws the data. One perspective is that when we fit the model by applying HTFA to a dataset, we “reverse” the generative process by starting with each subject’s data, which we use to estimate their subject-specific RBFs and weights, which we use in turn to estimate the global RBFs.

**Algorithm 1: HTFA’s generative process.** Here $F_s$ is the $K$ by $V_s$ factor image matrix for subject $s$, which depends on their subject-specific RBF centers ($\mu_i, \lambda_s$), RBF widths ($\lambda_1...K,s$), and their $V_s$ by 3 voxel location matrix ($R_s$).

\[
\begin{align*}
\text{for } k = 1 \text{ to } K & \text{ do } \\
& \text{Draw factor } k\text{'s global RBF center } \hat{\mu}_k \sim N(\mu_{\hat{\mu}_k}, \Sigma_{\hat{\mu}_k}); \\
& \text{Draw factor } k\text{'s global RBF width } \hat{\lambda}_k \sim N(\mu_{\hat{\lambda}_k}, \Sigma_{\hat{\lambda}_k}); \\
\end{align*}
\]

\[
\begin{align*}
\text{for } s = 1 \text{ to } S & \text{ do } \\
& \text{for } k = 1 \text{ to } K \text{ do } \\
& \text{Draw the } k^{th} \text{ factor's RBF center for subject } s: \\
& \mu_k,s \sim N(\hat{\mu}_k, \Sigma_{\mu_k}); \\
& \text{Draw the } k^{th} \text{ factor's RBF width for subject } s: \\
& \lambda_k,s \sim N(\hat{\lambda}_k, \Sigma_{\lambda_k}); \\
& \text{for } n = 1 \text{ to } N, \text{ do } \\
& \text{For image } n \text{ from subject } s, \text{ draw the } k^{th} \text{ factor's weight: } \\
& w_{n,k,s} \sim N(\mu_{w_{n,k}}, \Sigma_{w_{n,k}}); \\
& \text{end } \\
& \text{end } \\
& \text{for } n = 1 \text{ to } N, \text{ do } \\
& \text{Draw brain image } n \text{ for subject } s: \\
& y_{n,s} \sim N(\mu_{y_{n,s}}, \Sigma_{y_{n,s}}); \\
& \text{end } \\
\end{align*}
\]

Our goal in applying HTFA to a dataset is to infer the most probable values of the model’s hidden variables (e.g. RBF centers and widths, and factor weights). The posterior distribution $p(\Omega|Y_1...s)$ tells us how likely each hidden variable is to be set to a particular value, given the data and our prior assumptions about what these values should be. In theory, we could use Bayes’ rule to compute this posterior:
\[
p(\Omega|Y_1...s) = \frac{p(Y_1...s|\Omega)p(\Omega)}{p(Y_1...s)}. \]  
\[ \text{(Note that the hyperparameters have been omitted to simplify the notation.)} \]

\[
\text{However, computing the denominator (as for most models) is intractable:} \\
p(Y_1...s) = \int p(Y_1...s, \Omega)p(\Omega)d\Omega. \]  
\[ \text{(Note that these comparisons slightly underestimate the accuracy with which HTFA recovers the RBF centers and widths, since the original template image is never directly observed when we apply HTFA to a dataset (rather, only the subject data are observed; this is analogous to trying to estimate the mean of a Gaussian after observing several random draws).} \]

We found that the RBF centers recovered by applying HTFA to a subject data closely agree with the centers (Fig. 4A), and that the recovered RBF widths were reliably correlated with the template RBF widths (Fig. 4B).
HTFA tended to over-estimate the true RBF widths, and the degree to which a given RBF’s width was over-estimated was correlated with the Euclidean distance between the true and estimated RBF center \((r = 0.21, p = 0.03); \) Fig. 5. This positive correlation may be explained by the following intuition: if an RBF’s center is mis-estimated, then in order for the model to explain the variability in the data governed by the corresponding RBF in the template, HTFA compensates by growing the width of the estimated RBF to encompass the original (true) RBF.

We next asked whether HTFA accurately recovered the covariance structure in the synthetic data. We found that both the mean factor-to-factor connectivity matrix \((r = 0.84, p < 10^{-10}, \) Fig. 4C) and the mean across-image covariance matrix \((r = 0.96, p < 10^{-10}, \) Fig. 4D) recovered by HTFA were strongly correlated with the true covariance structure in the synthetic data. Taken together, these analyses show that HTFA is able to accurately infer (from synthetic data) the locations and sizes of the underlying RBF factors, the connectivity structure those factors exhibit, and the covariance structure across images. We next sought to evaluate a technique for automatically labeling the factors according to their response preferences and for using those labels to recover connectivity patterns in the data.

The factor weights in the synthetic dataset were determined according to the factors’ preferences (face-, scene-, left-, right-, or non-selective) and the experimental design matrix (which specified the category and location of the stimulus associated with each brain image). We estimated the response preferences of the factors that HTFA recovered by computing the correlations (across images) between each factor’s weights and each column of the design matrix (face, scene, left, right, and rest); this yielded a set of 5 correlations for each factor. We assigned each factor a response preference label according to which correlation was strongest (a factor was labeled as non-selective if the rest correlation was strongest, or if the absolute value of the strongest correlation, of the 5, was less than 0.01). As shown in the confusion matrix in Figure 6A, nearly all of the factors’ true preferences were recovered. We also examined the connectivity patterns (between the face-, scene-, left-, and right-selective factors) recovered by HTFA and found that they accurately reflected the connectivity patterns that we had injected in the synthetic data (Fig. 6B).

Our examinations of the synthetic dataset show that HTFA accurately recovers RBF centers and widths, per-image factor weights, and factor selectivity preferences (from synthetic data). This increases confidence in our general approach by showing that HTFA reliably resolves ambiguities in the data. We next turn to a series of analogous analyses on a real fMRI dataset.

**Case study 2: full-brain networks are modulated by stimulus location and category.** We examined an fMRI dataset that included data from 19 subjects who viewed synthetic (50% face, 50% scene) images on the left and right of the computer screen (Fig. 6B) and the mean across-image covariance matrix \((r = 0.96, p < 10^{-10}, \) Fig. 4D) recovered by HTFA. We applied HTFA to the dataset, using a cross-validation-based procedure to determine the optimal number of factors, \(K = 230\) (Fig. 7; also see Supporting information). As in our analyses of the synthetic dataset (Case study 1), we used the correlations between the factor weights (across images) and the experimental design matrices to label each factor as face-, scene-, left-, right-, or non-selective. We then examined how the average connectivity between these networks varied according to which aspects of the on-screen stimuli the subjects were attending to.

When we collapsed across the four experimental conditions (as in Fig. 6B), a clear pattern emerged. As shown in Figure 8, the connectivity between the category-matched and location-matched networks was reliably stronger than connectivity between the category-unmatched and location-matched networks \((t(18) = 2.34, p = 0.03)\). In other words, the location-matched network (e.g. the network of left-selective factors during face/left and scene/left trials) was more strongly connected with the category-matched network (e.g. face-selective factors during face/left and face/right trials) than with the category-unmatched network (e.g. scene-selective factors during face/left and face/right trials). In contrast, the location-unmatched network did not exhibit reliable differences in connectivity to the category-matched vs. unmatched networks \((t(18) = 0.0023, p = 0.99)\). These results indicate that full-brain networks are modulated by visual experience alone. Additional details may be found in the Supporting information.

**Discussion**

We proposed HTFA, a probabilistic approach to discovering and examining full-brain patterns of functional connectivity in multi-subject fMRI datasets. In Case study 1, we used a synthetic dataset to demonstrate HTFA’s ability to recover known patterns in the data, and in Case study 2 we used a real fMRI dataset to demonstrate how HTFA may be used to efficiently find and analyze full-brain connectivity patterns in real data.

**Related approaches.** Although most modern computer systems are capable of computing and storing (in RAM) a single subject’s voxel-to-voxel connectivity matrix, most commonly available systems cannot store many such matrices in RAM. Researchers have used several techniques to reduce the computational load. The most straightforward methods entail pre-selecting a small number of ROIs (e.g. motor cortex; [15]) or a seed voxel (e.g. a single voxel within the posterior cingulate; [16]). This reduces the connectivity matrix from a \(V \times V\) matrix (where, as above, \(V\) is the total number of voxels in each brain volume) to a much smaller \(V_{ROI} \times V\) matrix (where \(V_{ROI}\) is the number of seed or ROI voxels). However, reducing the connectivity matrix in this way precludes finding connectivity patterns unrelated to the ROI or seed region. For example, if the analysis is limited to connectivity patterns between the motor cortex and the rest of the brain, this precludes finding patterns of connectivity that do not involve the motor cortex (e.g. connectivity between prefrontal cortex and the hippocampus).

A related technique that does not require pre-selecting ROIs or seed regions is to compute the full voxel-to-voxel connectivity matrix, and then to threshold the connection strengths such that one only examines the most reliable connections [17]. This yields a sparse voxel-to-voxel connectivity matrix that may be easily manipulated (provided that it is sufficiently
One drawback to this approach is that it is not always clear how to set the connectivity strength threshold; for example, setting too high a threshold will leave out potentially important patterns, whereas setting too low a threshold will not substantially reduce the computational burden (as compared with examining the original voxel-to-voxel connectivity matrix). More generally, it is not clear that the strongest connections are necessarily the most informative; for example, a sub-threshold connection may carry information that could have been used to gain insights into the subject’s cognitive state.

Other approaches have focused on reducing the dimensionality of the connectivity patterns (see [18] for a comparison between thresholding-based and dimensionality-reduction-based approaches). For example, clustering-based approaches attempt to group together voxels that exhibit similar activation patterns over time, across subjects [19]. An alternative to clustering is to describe each image as a weighted sum of factors (where the number of factors is typically much smaller than the number of voxels in the original images). One may then examine connectivity patterns between the factors rather than between the voxels (this approach was originally developed in the positron emission tomography literature using PCA [20]).

HTFA takes a dimensionality reduction approach to examining full-brain connectivity patterns. Unlike standard approaches such as PCA and ICA, HTFA factors are constrained to be RBFs. Constraining HTFA factors to be RBFs has deeper geometric implications that we elaborate on in the next sub-section. Having RBF factors also makes HTFA especially well-suited to representing connectivity patterns, since each factor is spatially compact. In this respect, HTFA is qualitatively similar to spatial ICA [21]. However, unlike spatial ICA, HTFA further constrains factors to be in similar locations across subjects, providing a natural means of combining or comparing connectivity across individuals.

Benefits and costs of our approach. Our primary motivation in designing HTFA was to develop an efficient means of examining full-brain connectivity patterns in multi-subject datasets. What would it have taken to study the same patterns without RBFs? Constraining HTFA factors to be RBFs has important consequences in some applications. For example, suppose that (in some brain region), two subjects exhibit identical patterns of activity across images. Now suppose that we were to randomly perturb all of the odd voxels from the first subject and all of the even voxels from the second subject. Because voxel-based methods care only about the voxel labels rather than the voxel locations, perturbing the data in this way would perfectly disrupt the correlation between the subjects’ activity patterns in that brain region. In contrast, from the perspective of location-based models like HTFA, the subjects’ activation patterns would still look very similar. Although HTFA biases the subject-specific RBFs to be in similar locations, they are still allowed some flexibility in order to explain subject-specific idiosyncrasies (Fig. 1). Similar Intuitions have led researchers to apply spatial smoothing to fMRI data (typically by convolving the images with a Gaussian kernel). However, it is not always clear how much smoothing one should apply, nor is it clear that the amount of smoothing that would be “best” in one part of the brain would also be the best in another part of the brain. HTFA solves these issues by automatically determining the appropriate degree of effective smoothing in each part of the brain (via the RBF widths).

Another benefit of HTFA’s separation from voxel space is its ability to naturally fill in missing observations (a property we exploit to determine the optimal number of factors; Fig. 7). Techniques like probabilistic PCA [22] can fill in missing voxel activations using the data covariance matrix, provided that we observe at least some activations from those missing voxels (in other images). However, suppose that all activations from a given voxel were missing— or more realistically, suppose that we wish to estimate what the activations would have been at any arbitrary point in space. Because PCA does not explicitly represent the voxels in the spatial locations, probabilistic PCA can accurately predict activation patterns at these never-observed voxels. HTFA, by contrast, naturally predicts the missing data by simply evaluating each factor’s RBF at the corresponding location in space.

These missing data examples also provide insights into other benefits of allowing factors to exist in real space rather than considering only the set of voxel locations. For example, HTFA allows for different subjects’ data to be sampled at different resolutions, or to contain different numbers of voxels. In principle, although we have not explored this possibility formally in this paper, different subjects’ data may even come from different recording modalities (e.g., one subject may contribute fMRI data and another may contribute EEG data).

In this way, HTFA provides a common framework for describing neural data in general that transcends the specifics of the recording (modality, spatial or temporal resolution, etc.). For additional discussion of the benefits of spatial-based (rather than voxel-based) factors see [10], and for an example of how similar models may be used to analyze EEG data see [23].

We also note that using HTFA to examine connectivity patterns may not always out-perform voxel-based approaches. In particular, to the extent that the relevant patterns are high spatial frequency, those patterns will nearly always be better described by voxel-based approaches than RBF factors.
resenting brain images as sums of RBF factors effectively blurs out the images in space, where the amount of blurring is inversely proportional to the number of factors.) In theory we may optimize the number of factors to best describe the connectivity patterns in the dataset (see Supporting information and Fig. 7), although in practice we have found that high spatial frequency patterns are not typically recovered well by HTFA.

Concluding remarks. By providing a much more efficient means of examining full-brain functional connectivity than standard approaches, HTFA makes it possible to examine connectivity patterns using a wider range of techniques. The possibility of applying polynomial time and space algorithms (such as pattern classification algorithms) to connectivity data is particularly exciting.

Materials and methods

Applying HTFA to multi-subject fMRI datasets. We use a maximum a posteriori (MAP) estimation procedure to compute the most probable RBF factors and factor weights. The procedure has three basic steps: initialization (during which we set the prior over each factor), fitting subject-specific parameters for each subject (given the prior), and updating the global template (using the subject-specific parameters). When we carry out the full inference procedure, we iterate between updating the subject-specific parameters (using the current global template as the prior) and the global template (using the latest estimates of the subject-specific parameters) until the largest change in any parameters value from the previous iteration to the current iteration is less than a pre-determined threshold value, $\epsilon$. (We typically set $\epsilon$ to be the length of the longest voxel dimension.)

Each hidden variable, $x \in \Omega$, in the model (unshaded circles in Fig. 2) comes from a Gaussian with a mean parameter $\mu_x$ and either a covariance parameter $\Sigma_x$ (for the RBF centers) or variance parameter $\sigma^2_x$ (for the RBF widths and factor weights). We hold the covariance and variance parameters fixed (after initialization).

Once we initialize the prior, we use Algorithm S1 to generate a MAP estimate of the global RBF centers and widths, and the subject-specific RBF centers, RBF widths, and per-image factor weights. (The MAP estimates are the mean parameters of each hidden variable’s Gaussian.) At a descriptive level, the algorithm works by alternating between two steps: (1) update the subject-specific centers and widths given the global centers and widths, and (2) update the global centers and widths given the (new) subject-specific centers and widths. These two steps repeat until the global centers and widths stop changing (by more than a pre-defined threshold). To update the subject-specific centers and widths, we perform an additional two alternating steps, whereby in the first step we update the per-image factor weights (holding the centers and widths fixed) and in the second we update the centers and widths (holding the factor weights fixed). Once the global centers and widths have stopped changing, we run an additional step whereby we re-compute the per-image factor weights for each subject.

Estimating full-brain functional connectivity. The above parameter inference procedure yields, for each subject, an $N_s$ by $K$ matrix, $W_s$, of per-image factor weights. We can estimate the connectivity between each pair of factors by correlating the columns of $W_s$. We can also estimate the connectivity within a particular experimental condition by considering the pairwise correlations within only that subset of images (rows of $W_s$). This approach is analogous to standard voxel-wise techniques for estimating functional connectivity [15]. Further, because the columns of $W_{1:s}$ correspond to the same factors across the different subjects, since all of the factors are linked through the global template, the set of these weight matrices provide a convenient means of testing hypotheses related to the connectivity strengths.

Spatial attention dataset. The spatial attention dataset we examined in Case study 2 was collected as part of a separate study. A total of 19 subjects participated in the fMRI experiment. Each subject experienced two 3-minute localizer runs, four 5-minute task runs, and two 5-minute rest runs. During localizer runs (Fig. 9A), the subjects viewed images of faces and scenes, in either the upper left or the upper right side of the screen, while keeping their gaze fixed on a central point on the screen. Within a run, the spatial locations of the stimuli were held constant (i.e. stimuli were presented either on the right or the left within a given run), and the subjects viewed and made judgements about 12 face or scene stimuli in each of twelve 8 s alternative blocks (1.5 s per judgement; each block was separated by 18 s of rest. The orders in which subjects experienced the right and left localizer runs, and the category of the first stimulus in each run, were randomized across subjects.

During task runs (Fig. 9B), the subjects viewed composite stimuli. On every trial, the subject viewed two composite stimuli (one in the upper left and the other in the upper right of the screen) while keeping their gaze fixed on a central point on the screen. Each composite stimulus contained the average of a grayscale image of a face and a grayscale image of a scene. In different runs, the subjects were asked to attend to (and make male/female or indoor/outdoor judgements about) either the face or the scene components of either the left or the right stimulus. In this way, the subjects’ visual experiences were held roughly constant across different task runs, but the aspects of the stimuli they were instructed to attend to varied across runs. Finally, during rest runs, the subjects were instructed to lie quietly in the scanner with their eyes open (except while blinking as needed). Imaging parameters and image preprocessing methods may be found in the Supporting information.

ACKNOWLEDGMENTS. We acknowledge useful discussions with Jonathan Cohen, Justin Hubert, Talia Manning, Peter Ramadge, and Erez Simony. This work was supported by the NSF/NIH Collaborative Research in Computational Neuroscience Program, grant number NSF IIS-1009542. The content is solely the responsibility of the authors and does not necessarily represent the official views of our supporting organizations.

Fig. 1. Hierarchical Topographic Factor Analysis. **A.** A brain image and its associated reconstruction. The left sub-panel displays a single horizontal slice from a single subject; the right sub-panel displays its associated HTFA reconstruction, which we obtain by summing together the weighted images of the subject’s RBF factors. **B.** Explaining data across subjects. The left and middle sub-panels display example images from five subjects (left sub-panel), and their associated reconstructions (middle sub-panel). The right sub-panel displays the approximation of all of the single-subject images in the left sub-panel, obtained by setting the weights of the global template’s factors to their average weights in the subject-specific reconstructions. The locations of the RBFs in the global template reflect commonalities across subjects, whereas the single-subject RBF locations reflect the associated subject-specific idiosyncrasies.

Fig. 2. Graphical model for HTFA. Each variable in the model appears as a circle; hidden variables are unshaded and observed variables are shaded. Hyperparameters are denoted by dots. Arrows denote conditional dependence, originating at terms that appear on the right sides of conditionals and pointing towards terms that appear on the left sides. Rectangular plates denote repeated structure, where the number of copies is indicated within each plate (e.g., $N_s$, $S$, or $K$). For a comprehensive introduction to graphical models see [24]. Variables are defined in Algorithm 1 and in the text.
Fig. 3. Slices from an example synthetic image. Each panel displays a single slice along the z dimension of the $25 \times 25 \times 50$ voxel cube.
Fig. 4. Recovered structure from synthetic data. A. RBF centers. Each color denotes a different factor’s RBF in the synthetic template. The filled circles indicate the true locations of the RBF centers, and the open circles indicate the RBF centers inferred by HTFA. To facilitate visual comparison between the true and estimated locations, we have drawn a line between each recovered RBF center and the closest matching RBF center in the synthetic template. RBFs that have not been assigned any factors by HTFA (as determined by using this matching technique) are denoted by ×s. To fairly assess recovery of other parameters, in Panels B–C (and in our other analyses), we re-ordered the true factors to match the assignments shown in this Panel, and the factors that HTFA “missed” are not included in other comparisons or analyses. B. RBF widths. Each dot denotes an RBF’s true and estimated widths. The correlation reported in the panel is between the true and estimated RBF widths. C. Connectivity matrix. The panels display the true and estimated factor-to-factor connectivity matrices. To draw attention to the structure of these matrices and to facilitate visual comparison, we have re-ordered the rows and columns of these matrices so that factors with similar (true) preferences appear in adjacent rows and columns. The factor selectivities are shown along the left of the left panel: (face)-selective, (scene)-selective, (left)-selective, (right)-selective, and non-selective (X). D. Image covariance matrix. The panels display the true and estimated across-image covariance matrices.
Fig. 5. Center and width estimation errors are correlated. Each dot denotes a single recovered RBF. The center mismatch (x-axis) is measured as the Euclidean distance between the estimated RBF center and the nearest (true) RBF center in the synthetic template. The width mismatch (y-axis) is measured as the absolute difference between the estimated and true RBF width. The correlation reported in the panel is between the center and width mismatches.

Fig. 6. Recovering selectivity preferences. A. Confusion matrix. The confusion matrix displays the true and estimated selectivity preferences of the synthetic factors (relabeled as shown in Fig. 4A). The white cells along the right column display the percentages of estimated factors assigned to each category whose true labels were also in that category (hits; green) or whose true labels were not in that category (false alarms; red). The white cells along the bottom display the percentages of true factors that were assigned to each estimated category. The gray cell in the bottom right corner reports the average hit and false alarm rates across all categories. B. Connectivity between stimulus-selective and location-selective networks. The bars display the average estimated connectivity between each pair of stimulus- and location-selective networks, across all four simulated experimental conditions (face/left, face/right, scene/left, scene/right). The colored bars reflect connectivity for individual subjects, and the semi-transparent bars reflect the average (taken across subjects, ± SEM). The Category+ bars denote the matching category network (e.g., face networks during face/left or face/right trials) and the Category− bars denote opposite category networks. The Location+ and Location− bars follow the same naming convention. The comparison lines reflect t-tests between the corresponding bars (lower comparisons) and a t-test of the interactions (upper comparison). The ∗ ∗ ∗∗s indicate $p < 0.0001$. 
**Fig. 7.** Determining the optimal number of factors. We use a cross validation procedure to estimate, for each value of $K$, the model’s ability to predict connectivity patterns in held-out data. Each colored line displays the reconstruction errors as a function of $K$ for an individual subject in the Spatial attention dataset (Case study 2), and the gray ribbon displays the means of these curves ($\pm$ SEM) across subjects. The dotted black line denotes the optimal number of factors ($K = 230$). The full details of this approach may be found in the Supporting information.

**Fig. 8.** Network connectivity between category- and location-selective networks. This Figure is in the same format as Figure 6B. The bars denote the mean connectivity between the indicated pairs of networks, averaged across the four experimental conditions. The colored bars denote individual subjects’ connectivity patterns and the semi-transparent bars denote the averages across subjects. The error bars denote $\pm$ SEM. The * indicates $p < 0.05$. 
Fig. 9. Spatial attention task. A. Localizer run. During localizer runs, the subjects viewed either a face or a scene image, either in the upper left or the upper right of the screen, while keeping their gaze fixed on a central point (black ‘+’). B. Task run. During task runs, the subjects viewed composite stimuli on both sides of the screen while keeping their gaze fixed on a central point. On different runs, subjects made judgements about either the face or the scene component of either the left or the right stimulus.
Supporting Information

A probabilistic approach to full-brain functional connectivity analyses


*Princeton University, Princeton NJ, †New York University, ‡University of Guelph, and §Columbia University, New York NY

Submitted to Proceedings of the National Academy of Sciences of the United States of America

SI materials and methods

Initializing HTFA. The problem of finding the RBF centers, widths, and weights that best explain the data has many local optima. For example, if all of the RBFs are crowded into one small part of the brain, activity in that region might be explained well (provided that the factor weights were set appropriately), but activity in other brain regions would likely be explained poorly. However, if we were to move some of those RBFs into those poorly explained areas, then there is a tradeoff: we lose explanatory power in the region the RBFs were taken from, but we gain explanatory power in the region(s) the RBFs were moved to. That balance (i.e., whether it is better to keep the RBFs clustered in one small area or spread them out) may happen to tip in either direction. However, intuitively, since we are interested in approximating connectivity patterns throughout the entire brain, keeping the RBFs spread throughout the brain (rather than clustering them into one region) seems beneficial, unless we want to come across strong evidence to the contrary. By a similar token, keeping the RBFs relatively wide (rather than tiny) also seems beneficial, since large RBF widths mean that the factors will spread their mass over more brain area. We can bias the RBFs to spread throughout the brain and stay relatively wide by setting the prior distribution appropriately.

To set our prior over the global RBF centers, we select a subject, s, at random from the dataset. Subject s’s voxel locations are contained in the V_s by 3 matrix R_s. We use k-means clustering to group the voxel locations (i.e., rows of R_s) into K clusters (we describe how K is chosen below). We set the means of the prior over each global RBF’s center, \( \mu_{k,s} \), to one of these cluster means, ensuring that RBFs are (initially) spread throughout the regions of space occupied by the voxels. We set the prior covariances over the global RBF centers, \( \Sigma_s \), to the empirical covariance of R_s. Next, we set the covariances of the subject-specific RBF center distributions, \( \Sigma_{s,k} \), to \( \Sigma_s \). We set the means of the priors over each global RBF’s width, \( \mu_{k,s} \), to the length of the longest axis of subject s’s brain, and the prior variances over the global \( \Sigma_s \) and subject-specific (\( \Sigma_{s,k} \)) RBF widths to the variance in voxel locations along the longest axis of subject s’s brain. This biases the RBFs to be wide relative to subject s’s brain. Finally, we set the means of the priors over the per-image factor weights (\( \mu_{w_{1:s}} \)) to 0 and the variances of these priors (\( \sigma^2_{w_{1:s}} \)) to \( \frac{1}{2} \left( \max(Y) - \min(Y) \right) \). This biases the per-image factor weights towards 0, where the bias is (slightly) stronger for subjects who exhibit a wider range of voxel activations.

Optimizing the number of factors. For any K chosen in advance, we can use Algorithm S1 to estimate the K optimal RBF factors and factor weights. Given a dataset, how can we estimate the optimal value of K? Intuitively, if the images in a dataset are of low resolution, or if the patterns of voxel activations are relatively diffuse, then we would like K to be relatively small. On the other hand, for a high resolution dataset where the images tend to have very high spatial

frequency activation patterns, we would like K to be relatively large. In addition, our choice of K should be sensitive to overfitting; we should only choose a larger K if that allows the model to better generalize to previously unseen data. We use a cross-validation procedure to determine a value of K that satisfies the above desiderata.

We begin by specifying a set of possible values that K might reasonably take on. We have found the set \( K \in \{10, 20, 30, \ldots, 230, 240, 250\} \) to work well in practice. For each value of K in this set we choose (without replacement) 10 “training images” and 10 “test images” from each subject’s data. We also randomly label \( \frac{K}{10} \) of each subject’s voxels as “training voxels” and the other \( \frac{K}{10} \) as “test voxels.” Next, for each value of K, we use the inner while loop in Algorithm S1 to estimate (for each subject) their RBF centers and widths, using the training images. We then use the training voxels to estimate (again for each subject) the per-image factor weights in each of the test images (holding the RBF centers and widths fixed).

We can predict the activations of the held-out test voxels in the test images as follows

\[
\hat{Y}_{test}(K) = W_{test} F_{train}, \tag{S1}
\]

where \( Y_{test}(K) \) is the 10 by \( \frac{K}{10} \) matrix of predicted test voxel activations in the 10 test images, \( W_{test} \) is the 10 by K matrix of per-image factor weights, and \( F_{train} \) is the K by \( \frac{K}{10} \) matrix of factor images, constructed using RBFs parameterized by the centers and widths that were estimated using the training images and evaluated at the locations of the test voxels.

Since we are primarily interested in exploring connectivity patterns, we want to optimize K to best approximate the underlying voxel-to-voxel correlations matrix using the smallest number of factors. For each subject, and for each value of K, we can compare the true connectivity matrix for the held-out test voxels, \( \text{corr}(Y_{test}) \), to the inferred connectivity matrix, \( \text{corr}(\hat{Y}_{test}(K)) \). If we repeat this procedure \( m \) times per subject, then we obtain, for each value of K, a distribution of \( mS \) mean squared errors (between the corresponding entries of \( \text{corr}(Y_{test}) \) and \( \text{corr}(\hat{Y}_{test}(K)) \)).

In practice, through examining several fMRI datasets, we have found that the mean reconstruction errors tend to decrease monotonically with K (this trend seems to continue to
at least $K = 1000$), although the decreases often begin to flatten out as $K$ approaches (approximately) 100. This pattern may be seen in Figure 7, where we have plotted the mean squared reconstruction errors as a function of $K$ for each individual subject (colored lines) as well as the overall average (gray ribbon).

We can now select $K$ in one of two ways. The first approach is to select the minimum $K$ (from the set of possible values we chose at the beginning of the analysis) for which the means of the distributions of errors at that $K$ and the next highest value in the set are indistinguishable at a pre-selected significance level, $p_{thresh}$. While this approach is appealing in its simplicity, it is sensitive to “plateaus” in the reconstruction error curves. For example, in Figure 7, there is a small plateau between $K = 90$ and $K = 100$, after which the errors begin to decrease again as $K$ is increased. These plateaus can lead us to underestimate the optimal value of $K$ for our dataset.

Instead, we take an alternative approach by selecting the minimum $K$ for which the mean of the distribution of reconstruction errors is not distinguishable (via t-tests, at the significance threshold $p_{thresh}$) from the maximum value of $K$ we tested (e.g. $K = 250$). In this way, we are selecting a smaller value of $K$ than the maximum, that performs about as well as the highest value of $K$ in practice. Note that this approach can lead us to overestimate the optimal value of $K$ for our dataset; however, slightly overestimating $K$ is generally less problematic than underestimating it, as HTFA does not seem to suffer from overfitting (i.e., poor generalization to held-out data when $K$ gets too large) within the ranges of $K$ we are typically interested in. Nevertheless, choosing a more conservative $p_{thresh}$ will bias $K$ to be smaller, thereby somewhat mitigating the overestimation. In the spatial attention dataset we examined in Case study 2, we found that the mean of the distribution of reconstruction errors we observed using $K = 230$ factors was indistinguishable from that obtained using $K = 250$ factors (i.e., the maximum value of $K$ we examined) using $p_{thresh} = 0.001$. We therefore applied HTFA to the spatial attention dataset using $K = 230$ factors.

Imaging parameters. All subjects in the spatial attention dataset were scanned using a Siemens Skyra 3 T full-body scanner (Siemens, Erlangen, Germany) with a volume head coil. The functional runs comprised T2*-weighted gradient-echo-planar (EPI) sequences (voxel size = $3 \times 3 \times 3.5$ mm; repetition time [TR] = 1500 ms, echo time [TE] = 28 ms; flip angle = 64°; matrix = 64 × 64; slices = 27). We also collected, for each subject, a single high-resolution T1-weighted magnetization-prepared rapid-acquisition gradient echo (MPRAGE) image to facilitate registration and normalization (voxel size = 1 × 1 × 1 mm; TE = 3.3 ms; flip angle = 7°; matrix = 256 × 256; slices = 176), and a single fast low-angle shot (FLASH) field map to correct spatial distortions of the EPI images (vessel size = $0.75 \times 0.75 \times 3$ mm; TE = 2.6 ms; flip angle = 70°; matrix = 256 × 256; slices = 36).

Image preprocessing. We preprocessed the fMRI data using FEAT Expert (FSL, http://www.fmrib.ox.ac.uk/fsl). We removed the first six performance on real data.

We first generated a template subject (which all synthetic subjects would be based on) using 100 randomly generated RBFs. The RBF center parameters were selected by randomly choosing (without replacement) 100 of the voxel locations within the template subject’s image volume. We then chose the corresponding RBF widths by taking draws from a Gamma distribution with a shape parameter of 50 mm and a scale parameter of 33.33 mm. We randomly assigned factors to be face-selective, scene-selective, left-selective, right-selective, or non-selective. In other words, face- and scene-selective factors would increase their activity when the appropriate stimulus category was attended to (regardless of the stimulus location), and left- and right-selective factors would increase their activity when the appropriate location was attended to (regardless of the stimulus category). Non-selective factors would respond randomly throughout the experiment.

We next generated subject-specific RBF centers and widths by adding a small amount of Gaussian noise (drawn independently from $N(0,0.1)$) for every dimension of every center and for each width, for each factor and subject). The factor pref-
erences (i.e., face-, scene-, left-, right-, or non-selective) for each subject were set to be identical to the template factor preferences.

Finally, we selected per-image factor weights in each of 1728 images, for each simulated subject. These 1728 images included 288 images in each task block (face/left, face/right, scene/left, scene/right) 576 “rest” images (just as in the spatial attention fMRI dataset). We drew the factor weights (independently for each subject) in each task or rest block from multivariate Gaussians whose covariances were designed to reflect connectivity patterns between factors of different selectivities, depending on the experimental condition. In particular, for a given block (for a given subject), we constructed a 100 × 100 covariance matrix Σ_{task} whose entries were given by

\[
\Sigma_{task}(i, j) = \begin{cases} 
1 & : i = j \\
0.5 & : (i \land j) \land (i \neq j) \\
0 & : \text{otherwise}
\end{cases}
\]  

[S2]

where \((i \land j)\) denotes that factors \(i\) and \(j\) are both active during the given images, \((i \lor j)\) indicates that either \(i\) or \(j\) (or both) are active during the given images, and \(\neg(i \land j)\) indicates that either \(i\) or \(j\) (but not both) are active during the given images. Next, we ensure that Δ_{task} is positive definite. If \(A\) is a diagonal matrix whose entries are the eigenvalues of Σ_{task}, and \(B\) is a matrix whose columns are the corresponding eigenvectors, then we construct a new \(A\) as follows:

\[
\tilde{A}(i, i) = \begin{cases} 
10^{-15} & : A(i, i) < 0 \\
A(i, i) & : A(i, i) \geq 0
\end{cases}
\]  

[S3]

We can use \(\tilde{A}\) to construct a new covariance matrix, \(\tilde{\Sigma}_{task}\) that is similar to \(\Sigma_{task}\), but is guaranteed to be positive definite:

\[
\tilde{\Sigma}_{task} = B\tilde{A}B^T
\]  

[S4]

We also construct a mean vector \(\mu_{task}\), where \(\mu(i) = 1\) if factor \(i\) is active during the given images and is 0 otherwise. Finally, we draw each images’s factor weights in the next block of images from \(N(\mu_{task}, \tilde{\Sigma}_{task})\). The factor weights are drawn independently for each subject, such that the covariance structure is similar across subjects, but the specific patterns of activations are different.

Algorithm S1: MAP inference algorithm. The converged function returns true if all of the corresponding elements of its two arguments are within 1 voxel diameter (along the longest dimension), and false otherwise. The subsample function randomly selects (without replacement) y integer values between 1 and x. In each iteration of the inner while loop, we subsample α = 5000 voxels and β = 25 images from the given subject’s data, and update the subject-specific centers and widths using that subsample. The RBF function returns a matrix whose rows are radial basis functions evaluated as the given voxel locations, using each of the given parameters. We set the global iteration limit to ηglobal = 5 and the subject-specific iteration limit to ηsubj = 25. Note that using voxel and image subsampling and setting hard iteration limits makes this inference algorithm inexact; it is not guaranteed to converge to a local optimum. However, we have found it to perform well in practice, and these modifications allow the algorithm to quickly process very large datasets.

\[
\begin{align*}
& t \leftarrow 1; \\
& \{\mu_{i..K}(t), \lambda_{i..K}(t)\} \leftarrow \text{initPrior}(); \\
& \gamma_{\text{curr}} \leftarrow \{\mu_{i..K}(t), \lambda_{i..K}(t)\}; \\
& \gamma_{\text{prev}} \leftarrow \{\}; \\
& \text{while } \neg \text{converged}(\gamma_{\text{curr}}, \gamma_{\text{prev}}) \land t < \eta_{\text{global}} \text{ do} \\
& \quad \text{for } s = 1 \text{ to } S \text{ do} \\
& \quad \quad i \leftarrow 1; \\
& \quad \quad \phi_s \leftarrow \frac{N \cdot V_s}{M}; \\
& \quad \quad \psi_{\text{curr},s} \leftarrow \{\mu_{i..K,s}(t), \lambda_{i..K,s}(t)\}; \\
& \quad \quad \psi_{\text{prev},s} \leftarrow \{\}; \\
& \quad \quad \text{while } \neg \text{converged}(\psi_{\text{curr},s}, \psi_{\text{prev},s}) \land i < \eta_{\text{subj}} \text{ do} \\
& \quad \quad \quad \nu_{\text{sub}} \leftarrow \text{subsample}(V_s, \alpha); \\
& \quad \quad \quad \nu_{\text{sub}} \leftarrow \text{subsample}(N_s, \beta); \\
& \quad \quad \quad Y_{\text{sub}} \leftarrow Y_s(\nu_{\text{sub}}, \nu_{\text{sub}}); \\
& \quad \quad \quad R_{\text{sub}} \leftarrow R_s(\nu_{\text{sub}}); \\
& \quad \quad \quad F \leftarrow \text{RBF}(R_{\text{sub}}, \psi_{\text{curr},s}); \\
& \quad \quad \quad W \leftarrow \left(FF^T + \frac{1}{\sigma^2}I^K\right)^{-1}FY^T; \\
& \quad \quad \quad \psi_{\text{prev},s} \leftarrow \psi_{\text{curr},s}; \\
& \quad \quad \quad \psi_{\text{curr},s} \leftarrow \arg\min_{\mu_{i..K,s}, \lambda_{i..K,s}} \left(\psi_{\text{sub}} - WF^*\right)^2 + \frac{1}{2\sigma^2} \sum_{k=1}^{K} \left(\mu_{k,s} - \hat{\mu}_k(t)\right)^T \Sigma_{\mu}^{-1} \left(\mu_{k,s} - \hat{\mu}_k(t)\right) + \frac{1}{2\sigma^2} \sum_{k=1}^{K} \left(\lambda_{k,s} - \hat{\lambda}_k(t)\right)^2; \\
& \quad \quad \quad i \leftarrow i + 1; \\
& \quad \quad \text{end} \\
& \quad \quad \gamma_{\text{prev}} \leftarrow \gamma_{\text{curr}}; \\
& \quad \quad \text{for } k = 1 \text{ to } K \text{ do} \\
& \quad \quad \quad \hat{\mu}_k \leftarrow \left(\Sigma_{\hat{\mu}}^{-1} + \sum_{s=1}^{S} \Sigma_{\mu,s}\right)^{-1} \left(\Sigma_{\hat{\mu}}^{-1} \hat{\mu}_k + \sum_{s=1}^{S} \Sigma_{\mu,s} \mu_{k,s}\right); \\
& \quad \quad \quad \hat{\lambda}_k \leftarrow \left(\frac{1}{\lambda} + \sum_{s=1}^{S} \sigma_{\lambda,s}^2\right)^{-1} \left(\frac{1}{\lambda} \hat{\lambda}_k + \sum_{s=1}^{S} \frac{\lambda_{k,s}}{\sigma_{\lambda,s}^2}\right); \\
& \quad \quad \text{end} \\
& \quad \quad t \leftarrow t + 1; \\
& \quad \text{end} \\
& \quad \text{for } s = 1 \text{ to } S \text{ do} \\
& \quad \quad F_s \leftarrow \text{RBF}(R_s, \psi_{\text{curr},s}); \\
& \quad \quad W_s \leftarrow \left(FF^T + \frac{1}{\sigma^2}I^K\right)^{-1} F_s Y_s^T; \\
& \quad \text{end} \\
& \text{end} \\
& \text{end} \\
& \text{end}
\end{align*}
\]