Bayesian Structure Learning for Functional Neuroimaging

Mijung Park[*,1], **Oluwasanmi Koyejo**[*,1], **Joydeep Ghosh**[1],
Russell A. Poldrack[2], **Jonathan W. Pillow**[2]
[1]Electrical and Computer Engineering, [2]Psychology and Neurobiology
The University of Texas at Austin

Abstract

Predictive modeling of functional neuroimaging data has become an important tool for analyzing cognitive structures in the brain. Brain images are high-dimensional and exhibit large correlations, and imaging experiments provide a limited number of samples. Therefore, capturing the inherent statistical properties of the imaging data is critical for robust inference. Previous methods tackle this problem by exploiting either spatial sparsity or smoothness, which does not fully exploit the structure in the data. Here we develop a flexible, hierarchical model designed to simultaneously capture spatial block sparsity and smoothness in neuroimaging data. We exploit a function domain representation for the high-dimensional small-sample data and develop efficient inference, parameter estimation, and prediction procedures. Empirical results with simulated and real neuroimaging data suggest that simultaneously capturing the block sparsity and smoothness properties can significantly improve structure recovery and predictive modeling performance.

1 Introduction

Functional magnetic resonance imaging (fMRI) is an important tool for non-invasive study of brain activity. Most fMRI studies involve measurements of blood oxygenation (which is sensitive to the amount of local neuronal activity) while the participant is presented with a stimulus or cognitive task. Neuroimaging signals are then analyzed to identify the brain regions that exhibit a systematic response to the stimulation. This can be used to infer the functional properties of those brain regions. Estimating statistically consistent models for fMRI data is a challenging task. Typical experimental data consist of brain volumes represented by tens of thousands of noisy and highly correlated voxels, yet practical constraints generally limit the number of participants to fewer than 100 per experiment.

Predictive modeling (also known as "brain reading" or "reverse inference") has become an increasingly popular approach for studying fMRI data (Norman et al., 2006; Pereira et al., 2009; Poldrack, 2011). This approach involves decoding of the stimulus or task using features extracted from the neuroimaging data. Many different machine learning techniques have been applied to predictive modeling of fMRI data, including support vector machines (Cox, 2003), Gaussian naive Bayes (Mitchell et al., 2004) and neural networks (Hanson et al., 2004; Poldrack et al., 2009). The learned model parameters can also be used to infer associations between groups of voxels conditioned on the stimulus (Poldrack et al., 2009). Linear models are the preferred approach in this case, as the model weights are directly related to the image features (voxels). Interpretability and structure estimation are further simplified when the linear model returns sparse weights.

Various sparse regularizers have been applied to functional neuroimaging data to improve structure recovery (Carroll et al., 2009; Varoquaux et al., 2012). These models have had limited success due to the small number of samples and the high dimensions of the data. In particular, L1 regularized models typically select only a few features (voxels), and the selected subset of voxels can vary widely based on small changes in the hyperparameters or the data (Carroll et al., 2009). The high degree of correlation leads to further degeneration of the structure recovery and predictive performance. Similar empirical properties have been observed with other sparse modeling techniques

[*] M Park and O Koyejo contributed equally to this work.

Appearing in Proceedings of the 16th International Conference on Artificial Intelligence and Statistics (AISTATS) 2013, Scottsdale, AZ, USA. Volume 31 of JMLR: W&CP 31.

(Varoquaux et al., 2012). This observed behavior is consistent with the theoretical conditions for L1 regularized structure recovery (Zhao and Yu, 2006; Wainwright, 2009).

Here we show that statistical regularities in brain images can be exploited to improve estimation performance. Two properties are of particular interest: spatial block sparsity and spatial smoothness. Spatial sparsity results from the fact that the brain responds selectively, so that only small regions are activated during a particular task. Spatial smoothness, on the other hand, results from the fact that the brain regions activated extend across many (usually tens to hundreds of) voxels. Sparse blocks may not be located in close spatial proximity as different tasks or stimuli may be processed in very different brain regions (Poldrack, 2011). Sparse blocks may also be separated due to bilateral activation patterns for certain tasks. Much of the prior work in the domain of predictive modeling has focused on the sparse structure, whereas the spatial smoothness properties have mostly been ignored.

This paper introduces a novel prior distribution to simultaneously capture the spatial block sparsity and spatial smoothness structure of fMRI data. Our approach follows methods for structured predictive modeling using empirical Bayes (or maximum marginal likelihood) inference (e.g., Wipf and Nagarajan (2008); Sahani and Linden (2002); Park and Pillow (2011)). Our method builds directly on Automatic Locality Determination (ALD), which has a prior distribution that simultaneously captures sparsity and smoothness (Park and Pillow, 2011).

Our work differs from ALD in several respects: (i) we model several spatial clusters instead of a single spatial cluster; (ii) we apply the proposed prior model to both regression and classification problems; (iii) we propose an efficient representation to scale the model to high dimensional functional neuroimaging data.

The contributions of this paper are as follows:

- We propose a novel prior that simultaneously captures spatial block sparsity and smoothness.
- We develop efficient inference, parameter estimation, and prediction procedures for high-dimensional small-sample data.
- We present empirical results on simulated and real functional neuroimaging data. Our experiments show the effectiveness of our approach for predictive modeling and structure estimation.

We begin the discussion with an overview of the generative modeling approach in Section 2 and introduce the novel prior in Section 3. We discuss inference and parameter estimation applied to regression in Section 4 and classification in Section 5. Experimental results on synthetic and real brain data are presented in Section 6.

Notation : $\mathcal{N}\left(\mu, \sigma^2\right)$ represents a Gaussian distribution with mean μ and variance σ^2. We represent matrices by boldface capital letters and vectors by boldface small letters e.g. $\boldsymbol{M}, \boldsymbol{m}$ respectively. $\boldsymbol{M} = \mathrm{diag}(\boldsymbol{m})$ returns a diagonal $\boldsymbol{M}$ matrix with diagonal elements given by $\boldsymbol{M}_{i,i} = \boldsymbol{m}_i$. The determinant of a matrix $\boldsymbol{M}$ is given by $|\boldsymbol{M}|$, and $\mathrm{tr}(\boldsymbol{M})$ represents the trace of the matrix $\boldsymbol{M}$.

2 Generative model

We study whole brain images collected from subjects engaged in a controlled experiment. Let $\boldsymbol{x} \in \mathbb{R}^D$ be a feature vector representing the whole brain voxel activation levels collected into a D dimensional vector. The stimulus is represented by a variable y. This paper will focus on cases where y is real valued (regression), or y is discrete (classification). With N training examples, let $\boldsymbol{X} = [\boldsymbol{x}_1^\top|\boldsymbol{x}_2^\top|\ldots|\boldsymbol{x}_N^\top]^\top \in \mathbb{R}^{N\times D}$ represent the concatenated feature matrix, and let $\boldsymbol{\Theta}$ represent the model hyperparameters. Predictive modeling involves estimating the conditional distribution $p(y|\boldsymbol{x}, \mathcal{D}, \boldsymbol{\Theta})$ where data is denoted by $\mathcal{D}$.

We assume that the stimuli are generated from a hierarchical Bayesian model. Let the distribution of the stimuli be given by $p(y|\boldsymbol{w}, \boldsymbol{x}, \boldsymbol{\xi})$ where $\boldsymbol{\xi}$ are the likelihood model hyperparameters and $\boldsymbol{w} \in \mathbb{R}^D$ is a weight vector. The functional relationship between the voxel activations and the stimuli is assumed to be linear. The weights of this linear function are generated from a zero mean multivariate Gaussian distribution with covariance matrix $\boldsymbol{C} \in \mathbb{R}^{D\times D}$. The linear model and prior are given by:

$$f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x}, \quad p(\boldsymbol{w}|\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{0}, \boldsymbol{C}). \qquad (1)$$

where $\boldsymbol{\theta}$ represent hyperparameters that determine the covariance structure. We have suppressed the dependence of $\boldsymbol{C}$ on $\boldsymbol{\theta}$ to simplify the notation. Our objective is to parametrize this covariance matrix to capture prior smoothness and sparsity assumptions. We refer to this method as *Bayesian structure learning (BSL)*. Our approach consists of three main tasks:

1. Hyperparameter estimation using the parametric empirical Bayes approach.
2. Stimulus prediction for held-out images.
3. Structure estimation using a point estimate of weight vector.

Hyperparameter estimation: The set of model hyperparameters given by $\boldsymbol{\Theta} = \{\boldsymbol{\xi}, \boldsymbol{\theta}\}$ are learned using the parametric evidence optimization approach (Casella, 1985; Morris, 1983; Bishop, 2006). Evidence optimization (also known as *type-II* maximum likelihood), is a general procedure for estimating the parameters of the prior distribution in a hierarchical Bayesian model by maximizing the marginal likelihood of the observed data. We can compute the *evidence* by integrating out the model parameters $\boldsymbol{w}$ as:

$$p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\Theta}) = \int p(\boldsymbol{y}|\boldsymbol{w}, \boldsymbol{X}, \boldsymbol{\xi}) p(\boldsymbol{w}|\boldsymbol{\theta}) d\boldsymbol{w}.$$

The resulting maximizer is the maximum likelihood estimate $\hat{\boldsymbol{\Theta}}_{\text{ml}}$.

Stimulus prediction: The accuracy of the predictive model is estimated by computing predictions of held-out brain images. We estimate the predictive distribution of the target stimuli given by:

$$p(y_*|\boldsymbol{x}_*, \mathcal{D}, \boldsymbol{\Theta}) = \int p(y_*|\boldsymbol{x}_*, \boldsymbol{w}, \boldsymbol{\xi}) p(\boldsymbol{w}|\boldsymbol{\theta}, \mathcal{D}) d\boldsymbol{w} \quad (2)$$

where $p(\boldsymbol{w}|\boldsymbol{\theta}, \mathcal{D})$ is the posterior distribution of the parameters given the training data $\mathcal{D} = \{\boldsymbol{y}, \boldsymbol{X}\}$. The predictive distribution is applied to held-out brain images. Prediction performance provides evidence of accurate modeling and is generally useful for model validation.

Structure estimation: In addition to an accurate prediction of the stimuli, the weights of the linear mapping may be analyzed to infer stimulus dependent functional associations. This requires an appropriate point estimate. We compute the *maximum a posteriori* (MAP) estimate of the weight vector $\boldsymbol{w}$ by maximizing its (unnormalized) log posterior distribution conditioned on the estimated hyperparameters $\hat{\boldsymbol{\Theta}}_{\text{ml}}$. The estimated model parameter also specifies the recovered support. Ignoring constants independent of $\boldsymbol{w}$, the optimal parameter $\boldsymbol{w}_{\text{map}}$ is computed as the solution of:

$$\arg\min_{\boldsymbol{w}} \left[-\log p(\boldsymbol{y}|\boldsymbol{w}, \boldsymbol{X}, \boldsymbol{\xi}) + \tfrac{1}{2} \boldsymbol{w}^\top \boldsymbol{C}^{-1} \boldsymbol{w} \right]. \quad (3)$$

3 Prior covariance design

A smooth signal is characterized by its frequency content. In particular, the power of a smooth signal is concentrated near the zero frequency. We apply this intuition by designing a prior distribution that encourages low frequency weight vectors. Let $\boldsymbol{x} \in \mathbb{R}^D$ be the three dimensional tensor containing the brain volume where $D = D_x \times D_y \times D_z$. Each voxel is sampled on a regular three dimensional grid. Hence, we can measure the frequency content of $\boldsymbol{w} \in \mathbb{R}^D$ using the discrete Fourier transform (DFT) (Oppenheim and Schafer, 1989). Let $\mathsf{w} = \text{DFT}(\boldsymbol{w})$ represent the three dimensional discrete Fourier transform of $\boldsymbol{w}$ with the resulting discrete frequency spectrum $\mathsf{w} \in \mathbb{R}^D$. The weight vector $\boldsymbol{w}$ is considered smooth if the signal power of $\mathsf{w} = \text{DFT}(\boldsymbol{w})$ is concentrated near zero.

Let $\{\boldsymbol{e}_l \in \mathbb{R}^3\}$ represent the index locations in the frequency domain corresponding to the DFT of a three dimensional spatial signal i.e. $\boldsymbol{e}_l = \boldsymbol{0}$ corresponds to the zero frequency. As the signal is regularly sampled, $\boldsymbol{e}_l$ are on regular three dimensional grid. We encourage smooth weights with the use of a prior distribution $\mathsf{w} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{G})$. The prior covariance matrix $\boldsymbol{G} \in \mathbb{R}^{D \times D}$ is diagonal with entries:

$$\boldsymbol{G}_{l,l} = \exp\left(-\tfrac{1}{2} \boldsymbol{e}_l^\top \boldsymbol{\Psi}^{-1} \boldsymbol{e}_l - \rho \right),$$

where $\boldsymbol{\Psi} \in \mathbb{R}^{3 \times 3}$ is a diagonal scaling matrix and $\rho \in \mathbb{R}$ is a scaling parameter. The discrete Fourier transform of a real signal is symmetric around the origin (Oppenheim and Schafer, 1989). We use a diagonal scaling to ensure that this condition is satisfied. The result is an dimension-wise independent, symmetric prior distribution for w where the prior variance decreases exponentially in proportion to the Mahalanobis distance of the frequency index from the zero frequency.

The prior assumptions on the frequency domain signal w correspond to prior assumptions on the spatial weight vector $\boldsymbol{w}$ which can be recovered in closed form. Recall that the DFT is a linear operator (Oppenheim and Schafer, 1989). Let $\boldsymbol{B} \in \mathbb{R}^{D \times D}$ be the matrix representation of the the 3-dimensional discrete Fourier transform so $\mathsf{w} = \boldsymbol{B}\boldsymbol{w} = \text{DFT}(\boldsymbol{w})$. Similarly, the inverse 3-dimensional discrete Fourier transform (IDFT) operator is given by the Hermitian transpose of $\boldsymbol{B}$ so may compute $\boldsymbol{w} = \boldsymbol{B}^\top \mathsf{w} = \text{IDFT}(\mathsf{w})$. We can compute the marginal distribution of $\boldsymbol{w}$ by integrating out the prior $\mathsf{w} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{G})$. The resulting prior distribution on the spatial weight vector is given by:

$$\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{B}^\top \boldsymbol{G} \boldsymbol{B}).$$

Next, we augment the prior covariance matrix to capture the block spatial sparsity properties of the signal. Spatial blocks are modeled using a sum of C spatial clusters, where each cluster measures spatial locality. Let $\{\boldsymbol{z}_d\}$ represent the three dimensional sampling grid so each location d is associated with the corresponding voxel. Each cluster is defined by proximity to a central vector $\boldsymbol{\kappa}_c \in \mathbb{R}^3$. The intuition is that voxels near $\boldsymbol{\kappa}_c$ are considered active in the cluster c, while voxels far away are considered inactive. The sparsity promoting function for each cluster c at location d is given by:

$$s_c(d) = \gamma_c \exp\left(-\tfrac{1}{2} (\boldsymbol{z}_d - \boldsymbol{\kappa}_c)^\top \boldsymbol{\Omega}_c^{-1} (\boldsymbol{z}_d - \boldsymbol{\kappa}_c) \right),$$

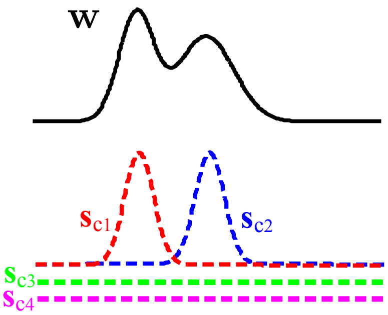

Figure 1: **Visualization of spatial block sparsity.** An example of 1-dimensional weight vector $\boldsymbol{w}$ with estimated spatial prior clusters $\{\boldsymbol{s}_c\}$. We used four clusters in the prior. Two of them $\{\boldsymbol{s}_{c1}, \boldsymbol{s}_{c2}\}$ specify the support of $\boldsymbol{w}$, and the rest $\{\boldsymbol{s}_{c3}, \boldsymbol{s}_{c4}\}$ were pruned out.

where $\boldsymbol{\Omega}_c \in \mathbb{R}^{3\times 3}$ is a symmetric positive definite matrix and $\gamma_c \in \mathbb{R}_+$ is a positive weight. The sparsity promoting functions are collected into a vector $\boldsymbol{s}_c \in \mathbb{R}^D$.

The clusters are accumulated into a single spatial sparsity promoting function:

$$\begin{aligned} s(d) &= \sum_{c=1}^{C} s_c(d) \\ &= \sum_{c=1}^{C} \gamma_c \exp\left(-\tfrac{1}{2}(\boldsymbol{z}_d - \boldsymbol{\kappa}_c)^\top \boldsymbol{\Omega}_c^{-1}(\boldsymbol{z}_d - \boldsymbol{\kappa}_c)\right), \end{aligned}$$

and collected into a vector $\boldsymbol{s} \in \mathbb{R}^D$. This modeling approach allows us to capture arbitrarily shaped blocks as a weighted sum of the elliptical clusters. Blocks that are not utilized can be identified as blocks with $\gamma_c = 0$ and pruned. Hence C is an upper bound on the number of spatial blocks explicitly captured by the prior. Collectively, $\{\boldsymbol{s}_c\}$ select the support of $\boldsymbol{w}$ (see Fig. 1). The spatial cluster centers $\boldsymbol{\kappa}_c$ are constrained by the boundaries of the cuboid. We also set $s(d) = 0$ for all voxels outside the brain volume. This will ensure that the estimated weight vector corresponding to these voxels remains zero.

We now combine the spatial sparsity promoting functions with the prior covariance matrix for spatial smoothness. We define a diagonal matrix $\boldsymbol{S} = \mathrm{diag}(\boldsymbol{s}^{\frac{1}{2}}) \in \mathbb{R}^{D\times D}$ that imposes locality in space on the prior covariance. The modified prior covariance matrix $\boldsymbol{C} \in \mathbb{R}^{D\times D}$ is now given by:

$$\boldsymbol{C} = \boldsymbol{S}\boldsymbol{B}^\top \boldsymbol{G}\boldsymbol{B}\boldsymbol{S}. \tag{4}$$

Our proposed design combines the notions of spatial block sparsity and spatial smoothness into a single prior covariance matrix. With a fixed number of clusters C, the covariance matrix is defined by the hyperparameters $\boldsymbol{\theta} = \{\boldsymbol{\Psi}, \rho, \{\gamma_c, \boldsymbol{\kappa}_c, \boldsymbol{\Omega}_c\}_{c=1}^{C}\}$.

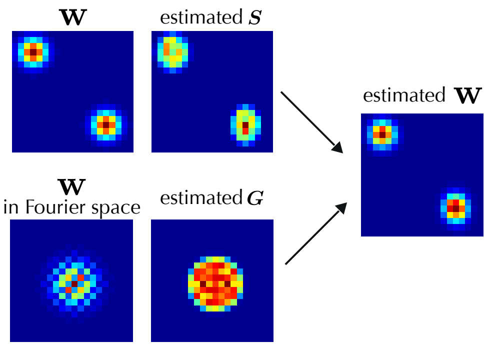

Figure 2: **Combining spatial block sparsity and spatial smoothness**. *Top*: The true 2-dimensional weight vector $\boldsymbol{w}$ (left) and the estimated spatial block sparsity matrix $\boldsymbol{S}$ (right). *Bottom*: $\mathsf{w} = \mathrm{DFT}(\boldsymbol{w})$ in Fourier space (left) and the estimated frequency sparse prior variance $\boldsymbol{G}$ (right). *Right*: The estimated weight vector $\boldsymbol{w}_{\mathrm{map}}$ using the proposed prior covariance.

The support of the weight vector $\boldsymbol{w}$ is determined by the structure of the covariance matrix $\boldsymbol{C}$ through the sparsity of the matrix $\boldsymbol{S}$. Elements of the weight vector $\boldsymbol{w}$ with zero prior covariance will remain sparse. To illustrate this effect, suppose the diagonals of $\boldsymbol{S}$ contain t non-zero elements, then the rows and columns of $\boldsymbol{C}$ corresponding to the $u = D - t$ sparse indexes are zero. Without loss of generality, there exists a permutation matrix $\boldsymbol{P} \in \mathbb{R}^{D\times D}$ such that the covariance matrix can be partitioned as:

$$\boldsymbol{P}^\top \boldsymbol{C}\boldsymbol{P} = \left(\begin{array}{c|c} \tilde{\boldsymbol{C}} & \boldsymbol{0}_{t\times u} \\ \hline \boldsymbol{0}_{u\times t} & \boldsymbol{0}_{u\times u} \end{array}\right),$$

where $\tilde{\boldsymbol{C}} \in \mathbb{R}^{t\times t}$ is the non-zero sub-matrix of $\boldsymbol{C}$, and $\boldsymbol{0}$ are all zero matrices of the appropriate size. Hence, the Gaussian prior on $\tilde{\boldsymbol{w}} = \boldsymbol{P}^\top \boldsymbol{w}$ used to compute the MAP estimate Eq. 3 is given as:

$$\tilde{\boldsymbol{w}}^\top (\boldsymbol{P}^\top \boldsymbol{C}\boldsymbol{P})^{-1} \tilde{\boldsymbol{w}} = \tilde{\boldsymbol{w}}^\top \left(\begin{array}{c|c} \tilde{\boldsymbol{C}}^{-1} & \boldsymbol{0}_{t\times u} \\ \hline \boldsymbol{0}_{u\times t} & \boldsymbol{0}_{u\times u}^{-1} \end{array}\right) \tilde{\boldsymbol{w}}.$$

The prior evaluates to infinity unless $\boldsymbol{w} = 0$ for all indexes corresponding to the zero entries in the diagonal of $\boldsymbol{S}$. In practice, this can be implemented by pruning all the dimensions of $\mathbf{w}$ corresponding to zero spatial weights before the MAP estimation procedure.

Efficient implementation of $\boldsymbol{B}$ operator: Although the discrete Fourier transpose matrix $\boldsymbol{B}$ is a structured matrix, its storage storage costs are of order $\mathcal{O}(D^2)$, and transformation to the frequency domain using a matrix vector product has a computational cost of costs $\mathcal{O}(D^2)$. This cost may be prohibitive as the DFT transformation is utilized in the inner loop of the evidence optimization and point estimation of the weight vector. These costs can be significantly reduced by exploiting the equivalence between the $\boldsymbol{B}$ and the three dimensional discrete Fourier transform DFT$(\cdot)$. Using this approach, $\boldsymbol{B}$ incurs no storage costs, and transformation to the frequency domain requires computation costs of $\mathcal{O}(D \log D)$. For instance, the covariance matrix is involved in hyperparameter estimation via quadratic terms of the form $\boldsymbol{U}^\top \boldsymbol{C} \boldsymbol{V} = \boldsymbol{U}^\top \boldsymbol{S} \boldsymbol{B}^\top \boldsymbol{G} \boldsymbol{B} \boldsymbol{S} \boldsymbol{V}$. This can implemented by (i) spatial scaling: spatial scaling: $\boldsymbol{u} = \text{diag}(\boldsymbol{S}) \odot \boldsymbol{U}$ and $\boldsymbol{v} = \text{diag}(\boldsymbol{S}) \odot \boldsymbol{V}$, (ii) discrete Fourier transform: $\mathsf{u} = \text{DFT}(\boldsymbol{u})$ and $\mathsf{v} = \text{DFT}(\boldsymbol{v})$, and (iii) weighted inner product: $\mathsf{u}^\top \boldsymbol{G} \mathsf{v}$, where $\text{diag}(\boldsymbol{S}) \odot \boldsymbol{U} = \boldsymbol{S}\boldsymbol{U}$ corresponds to the product of each element of $\text{diag}(\boldsymbol{S})$ with the corresponding row of $\boldsymbol{U}$. The result can be computed even more efficiently by using recent algorithms for sparse fast Fourier transforms (Hassanieh et al., 2012), exploiting the frequency sparsity recovered by the covariance matrix. This further extension is left for future work.

3.1 Efficient representation with high dimensions

In this high dimensional scenario, the size of $\boldsymbol{w}$ and $\boldsymbol{C}$ render naïve implementation computationally infeasible. On the other hand, typical neuroimaging datasets contain a relatively small number of samples. We exploit this small sample property to improve the computational efficiency of representation and evidence optimization. Recall that that the relationship between the voxel response and the stimulus is given by the linear function $f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x}$ and the weight vector is drawn from a Gaussian distribution Eq. 1. Let $\boldsymbol{f} = [f(\boldsymbol{x}_1), f(\boldsymbol{x}_2), \ldots, f(\boldsymbol{x}_N)]^\top \in \mathbb{R}^N$. The prior distribution of $\boldsymbol{f}$ can be recovered in closed form by integrating out the weight vector. This results in an equivalent representation of the generative model in the *function space*:

$$\boldsymbol{f} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K}), \quad \boldsymbol{K} = \boldsymbol{X}\boldsymbol{C}\boldsymbol{X}^\top. \tag{5}$$

The reader may notice the similarity to the Gaussian process prior (c.f. chapter 2.1 of Rasmussen and Williams (2005)). In fact, Eq. 5 is equivalent to a Gaussian process prior over linear functions with mean 0 and covariance function $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^\top \boldsymbol{C} \boldsymbol{x}_j$. This function space representation significantly reduces the complexity of inference when $N \ll D$. For instance, the computational complexity of inference in regression is reduced from $\mathcal{O}(D^3)$ to $\mathcal{O}(N^3)$, and storage requirements for the covariance can be reduced from $\mathcal{O}(D^2)$ to $\mathcal{O}(N^2)$.

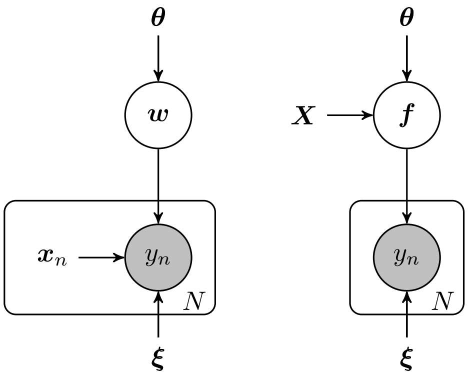

Figure 3: Equivalent representations of generative model in the weight space (left) and the dual function space (right). $\boldsymbol{\theta}$ are parameters of the prior distribution, and $\boldsymbol{\xi}$ are likelihood model parameters.

4 BSL for regression

The continuous valued stimuli are modeled as independent Gaussian distributed variables $p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\xi}) = \mathcal{N}(f(\boldsymbol{x}), \sigma^2)$. Without loss of generality, we will assume that the data is normalized so the stimuli are zero mean. Hence, the likelihood hyperparameters $\boldsymbol{\xi}$ represent the observed noise variance σ^2. Let $\boldsymbol{y} = [y_1, y_2, \ldots, y_N]^\top \in \mathbb{R}^N$ represent the N training stimuli collected into a vector. In this section, we summarize the procedures for evidence optimization, stimuli prediction and weight estimation.

4.1 Hyperparameter estimation

As the prior distribution and the likelihood are both Gaussian, the prior Eq. 5 can be integrated out in closed form. The result is the evidence:

$$p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\Theta}) = \mathcal{N}(\boldsymbol{0}, \boldsymbol{K}_y).$$

where $\boldsymbol{K}_y = \boldsymbol{K} + \sigma^2 \mathbf{I}$. We estimate the model hyperparameters by maximizing the corresponding marginal log likelihood which is given by:

$$\log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\Theta}) = -\frac{1}{2} \boldsymbol{y}^\top \boldsymbol{K}_y^{-1} \boldsymbol{y} - \frac{1}{2} \log |\boldsymbol{K}_y| - \frac{N}{2} \log 2\pi.$$

The log evidence can be optimized efficiently using gradient based direct optimization techniques (Rasmussen and Williams, 2005).

4.2 Predictive distribution

The predictive distribution is computed by marginalizing out the model parameters with respect to their posterior distribution. The posterior distribution of the noise free response $f_* = f(\boldsymbol{x}_*)$ can be computed in closed form (Rasmussen and Williams, 2005) as:

$$p(f_*|\boldsymbol{x}_*, \mathcal{D}) = \mathcal{N}(\mu, \Sigma) \tag{6}$$
$$\mu = \boldsymbol{x}_*^\top \boldsymbol{C}\boldsymbol{X}^\top(\boldsymbol{K} + \sigma^2\mathbf{I})^{-1}\mathbf{y}$$
$$\Sigma = \boldsymbol{x}_*^\top \boldsymbol{C}\boldsymbol{x}_* + \boldsymbol{x}_*^\top \boldsymbol{C}\boldsymbol{X}^\top(\boldsymbol{K} + \sigma^2\mathbf{I})^{-1}\boldsymbol{X}\boldsymbol{C}\boldsymbol{x}_*,$$

where $\mathcal{D} = \{\boldsymbol{y}, \boldsymbol{X}\}$ represents the training data. We use the mean of the posterior distribution a point estimate of the model prediction.

4.3 Point estimate of weight vector

Given the trained hyperparameters, the point estimate that maximizes the posterior distribution is equal to the posterior mean of the weight vector. The posterior distribution of the weight vector can be computed in closed form as:

$$p(\boldsymbol{w}|\mathcal{D}, \boldsymbol{\Theta}) = \mathcal{N}(\sigma^2 \boldsymbol{\Sigma}\boldsymbol{X}^\top\boldsymbol{y}, \boldsymbol{\Sigma}) \tag{7}$$

where $\boldsymbol{\Sigma} = (\boldsymbol{C}^{-1} + \sigma^2\boldsymbol{X}^\top\boldsymbol{X})^{-1}$. Note that only the mean of the posterior distribution is required for the point estimate. Yet this closed form may be computationally infeasible with high dimensional data. A scalable alternative approach is direct maximization of the (unnormalized) posterior distribution as described in Eq. 3. Ignoring constant terms, the resulting optimization is given by:

$$\boldsymbol{w}_{\text{map}} = \arg\min_{\boldsymbol{w}} \left[\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|_2^2 + \sigma^2\boldsymbol{w}^\top\boldsymbol{C}^{-1}\boldsymbol{w}\right]. \tag{8}$$

This is a regularized least squares problem and can be solved efficiently using standard optimization techniques.

5 BSL for classification

We employ a classification approach when the target stimuli consists of a set of discrete items. Let J be the total number of stimuli classes and let y_n^j be an indicator variable with $y_n^j = 1$ if the n^{th} image is from class j, and $y_n^j = 0$ otherwise. These are collected into the vector $\boldsymbol{y}^j = [y_1^j, \ldots, y_N^j]^\top$, and the combined stimuli is given by $\boldsymbol{y} = [(\boldsymbol{y}^1)^\top, \ldots, (\boldsymbol{y}^J)^\top]^\top$. We use a separate linear function for each class so the resulting weights can be interpreted directly as a discriminative stimulus signature.

The linear function response for each class is computed as $\boldsymbol{f}^j = [f_1^j, \ldots, f_N^j]^\top \in \mathbb{R}^N$ where $f_n^j = f^j(\boldsymbol{x}_n)$ and the combined function vector is given by $\boldsymbol{f} = [(\boldsymbol{f}^1)^\top, \ldots, (\boldsymbol{f}^J)^\top]^\top$. Each class function is drawn from a multivariate Gaussian prior $\boldsymbol{f}^j \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K}^j)$ a with a class specific covariance $\boldsymbol{K}^j = \boldsymbol{X}\boldsymbol{C}^j\boldsymbol{X}^\top$ as described in Eq. 5. We assume that the prior distributions of the class functions are uncorrelated, Hence, we can define the prior distribution of the combined vector as $\boldsymbol{f} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K})$, where $\mathbf{K}$ is a block diagonal matrix with blocks $\boldsymbol{K}^j$.

The probability of n^{th} stimulus belonging to j^{th} class class is defined by the softmax:

$$p(y_n^j|\{f_n^j\}_{j=1}^J) = \pi_n^j = \frac{\exp(f_n^j)}{\sum_{l=1}^J \exp(f_n^l)}. \tag{9}$$

Assuming that each of the N targets $\{y_n^j\}_{j=1}^J$ are conditionally independent, the log-likelihood of the data is given by:

$$\mathcal{L}(\boldsymbol{f}) = \log p(\boldsymbol{y}|\boldsymbol{f}) = \boldsymbol{y}^T\boldsymbol{f} - \sum_{n=1}^N \log\left(\sum_{l=1}^J \exp(f_n^l)\right). \tag{10}$$

5.1 Hyperparameter estimation

The evidence function is not available in closed form. We employ an approximate evidence approach based on an approximate posterior. The posterior distribution of the latent functions do not have a closed form expression. We estimate an approximate posterior distribution using the Laplace approximation (Rasmussen and Williams, 2005; Park et al., 2011) based on a Gaussian approximation to the posterior distribution at the mode. The approximate posterior takes the form:

$$p(\boldsymbol{f}|\mathcal{D}) \approx \mathcal{N}(\boldsymbol{f}_{\text{map}}, \boldsymbol{\Lambda}^{-1}) \tag{11}$$

where $\boldsymbol{f}_{\text{map}}$ is the MAP parameter estimate. The $\boldsymbol{f}_{\text{map}}$ is computed by maximizing the unnormalized log-posterior:

$$\Phi(\boldsymbol{f}) = \mathcal{L}(\boldsymbol{f}) - \frac{1}{2}\boldsymbol{f}^T\boldsymbol{K}^{-1}\boldsymbol{f} - \frac{1}{2}\log|2\pi\boldsymbol{K}|. \tag{12}$$

The posterior covariance is given by the Hessian at $\boldsymbol{f}_{\text{map}}$ computed as:

$$\boldsymbol{\Lambda}^{-1} = \nabla\nabla\Phi(\boldsymbol{f}) = -\boldsymbol{K}^{-1} - \boldsymbol{H}.$$

where $\boldsymbol{H} = -\frac{\partial^2}{\partial \boldsymbol{f}^2}\mathcal{L}(\boldsymbol{f}) = \text{diag}(\boldsymbol{\pi}) - \Pi\Pi^T$, and Π is the matrix of size $Jn \times n$ formed by vertically stacking the matrices $\text{diag}(\boldsymbol{\pi}^j)$.

Finally, we optimize the evidence at $\boldsymbol{f} = \boldsymbol{f}_{\text{map}}$. This is given by:

$$p(\boldsymbol{y}|\boldsymbol{\theta}) = \frac{p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{\theta})}{p(\boldsymbol{f}|\boldsymbol{y}, \boldsymbol{\theta})} \approx \frac{\exp\left(\mathcal{L}(\boldsymbol{f})\right)\mathcal{N}(\boldsymbol{f}|0, \boldsymbol{K})}{cN(\boldsymbol{f}|\boldsymbol{f}_{\text{map}}, \boldsymbol{\Lambda}^{-1})}.$$

The resulting evidence optimization follows the approach outlined in Rasmussen and Williams (2005).

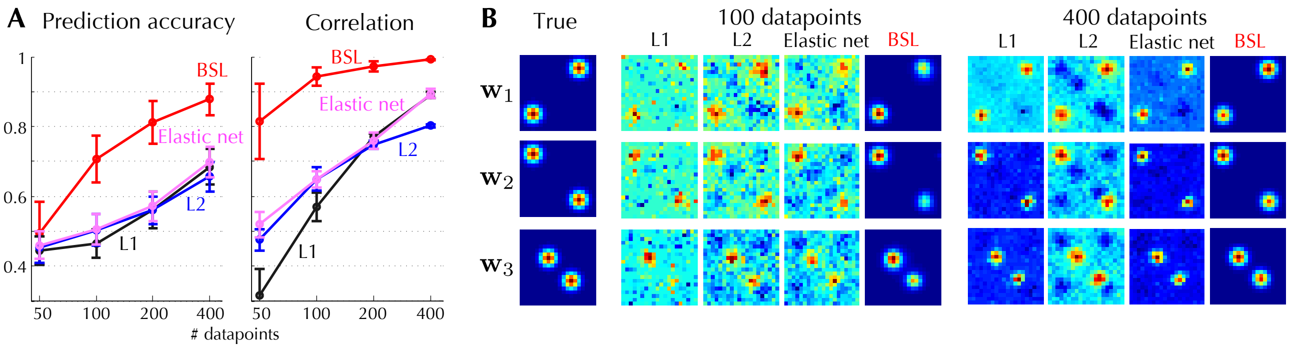

Figure 4: **2D simulated example for 3-class classification** *A*: Prediction accuracy, and correlation between true weight vectors and estimates obtained by L1, L2, elastic net regularization methods, and BSL. *B*: True weight vectors for each class, and the estimates obtained by each method using 100 and 400 training data points. BSL outperforms L1, L2 and elastic net regularized models both in terms of classification accuracy and support recovery.

5.2 Predictive distribution

The posterior predictive distribution is analytically intractable, so employ the approximate Gaussian posterior Eq. 11. We approximate the posterior predictive distribution for class j as:

$$\begin{aligned} p(f_*^j|\boldsymbol{x}_*, \mathcal{D}) &= \mathcal{N}(\mu, \Sigma) \qquad (13)\\ \mu &= \boldsymbol{x}_*^\top \boldsymbol{C}^j \boldsymbol{X}^\top (\boldsymbol{K}^j)^{-1} \boldsymbol{f}^j_{\text{map}}\\ \Sigma &= \text{diag}(\boldsymbol{k}(\boldsymbol{x}_*, \boldsymbol{x}_*)) - \boldsymbol{Q}_*^\top (\boldsymbol{K} + \boldsymbol{H}^{-1})^{-1} \boldsymbol{Q}_*, \end{aligned}$$

where $\boldsymbol{k}(\boldsymbol{x}_*, \boldsymbol{x}_*)$ is the vector of covariances with the j^{th} element given by $\boldsymbol{x}_*^\top \boldsymbol{C} \boldsymbol{x}_*$ and $\boldsymbol{Q}_*$ is the $(JN \times J)$ matrix:

$$\boldsymbol{Q}_* = \begin{pmatrix} \boldsymbol{X}^\top \boldsymbol{C}^1 \boldsymbol{x}_*^\top & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \boldsymbol{X}^\top \boldsymbol{C}^2 \boldsymbol{x}_*^\top & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{X}^\top \boldsymbol{C}^J \boldsymbol{x}_*^\top \end{pmatrix}.$$

Given the predictive distribution Eq. 13, the predictive class probabilities can be computed using the Monte Carlo sampling approach as shown in (Rasmussen and Williams, 2005).

5.3 Point estimate of weight vector

The point estimate of the weight vector is computed as the vector that maximizes the unnormalized log posterior distribution $\log p(\boldsymbol{w}|\mathcal{D}, \boldsymbol{\Theta})$. Ignoring constant terms, the resulting $\boldsymbol{w}_{\text{map}} \in \mathbb{R}^{DJ}$ is given by:

$$\arg\min_{\boldsymbol{w}} \left[\boldsymbol{y}^T \boldsymbol{f} - \sum_{n=1}^{N} \log \left(\sum_{l=1}^{J} \exp(f_n^l) \right) + \boldsymbol{w}^\top \boldsymbol{C}^{-1} \boldsymbol{w} \right]. \qquad (14)$$

We have retained the linear function representation of the likelihood for compactness, however, note that the optimization problem is posed in the weight space. This optimization corresponds to a regularized generalized linear model and can be solved efficiently using standard optimization techniques.

6 Experimental results

We present experimental results comparing the proposed Bayesian structured learning (BSL) model to regularized generalized linear models for predictive modeling.

Simulated data

We first tested our method on simulated data in a 3-class classification setting. We generated N random 2-dimensional images where each pixel was generated independently from the standard normal distribution $\mathcal{N}(0, 1)$. We also generated a set of weight vectors as shown in Fig. 4B (first column). The stimuli responses were generated using a multinomial distribution Eq. 9 and hard thresholded into one of three classes. The dimensionality of each weight vector was 20 by 20, resulting in a $D = 1200$ parameter space. We first examined the prediction accuracy of estimates obtained by L1, L2, elastic net regularization (Zou and Hastie, 2005), and our method (BSL). The average prediction accuracy (from 10 independent repetitions) is shown in Fig. 4A (left) as a function of the number of training samples. The estimated weight vectors from each method are shown in Fig. 4B, (right) using 100 and 400 data points, respectively. We computed the correlation coefficients between the true weight vector and estimates obtained by each method to test the support recovery performance. These are shown in Fig. 4A, (right). As shown in the presented results, our method outperforms other methods in terms of prediction ac-

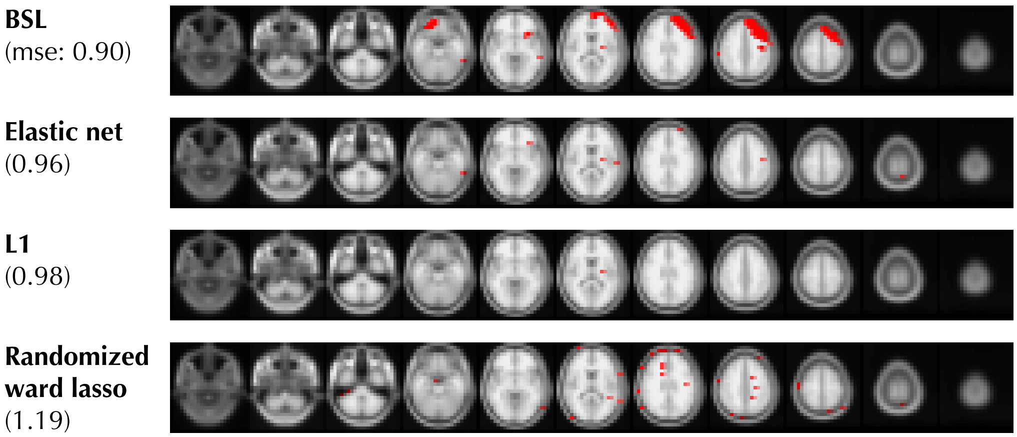

Figure 5: **Support (in red) of the estimated weights from each method using real fMRI data.** Each row shows slices of the brain from the top of the skull. The magnitude of the weight vector is not shown. *First row*: Estimate obtained by BSL. *Second row*: Estimate obtained by elastic net regularization. *Third row*: Estimate obtained by L1 regularization. *Fourth row*: Estimate obtained by randomized ward lasso. Numbers in parenthesis are the 10-fold cross validation average mean squared error from each method. BSL outperforms other methods in terms of mean squared error and recovers an interpretable support.

curacy as well as support recovery.

Functional neuroimaging data

fMRI data were collected from 126 participants while the subjects performed a stop-signal task (Aron and Poldrack, 2006). For each subject, contrast images were computed for "go" trials and successful "stop" trials using a general linear model with FMRIB Software Library (FSL), and these contrast images were used for regression against estimated stop-signal reaction times. The fMRI data is was down-sampled to $22 \times 27 \times 22$ voxels using the `flirt applyXfm` tool (Alpert et al., 1996).

Fig. 5 shows the recovered support from the proposed BSL, L1 regularized regression, elastic net regularized regression, and randomized ward lasso using hierarchical spatial clustering (Varoquaux et al., 2012). We tested each method using 10-fold cross-validation and computed the mean square error (MSE) performance averaged over the 10 folds. The hyperparameters for L1, L2, elastic net, and randomized ward lasso were computed using an inner cross-validation loop. In BSL, we initialized the hyperparameters from the L2 estimate to avoid some of the issues with local minima. For spatial sparsity, 20 clusters were assumed based on domain expertise, and unused blocks were pruned out automatically during the hyperparameter estimation.

The results from L2 regularized regression are not shown as the returned weights had full support, hence direct interpretation of the weight vector was infeasible. In addition to the presented results, we tested the relevance vector machine (Tipping, 2001) (6.6%) and stability selection lasso (Meinshausen and Bhlmann, 2010) (6.7%), relative increase in MSE compared to BSL are given in parenthesis. The corresponding images are not shown due to space constraints. We also tested the special cases of BSL with block sparsity alone (2.6%) and spatial correlation alone (3.2%).

The regions identified by BSL encompass a set of regions (including right prefrontal cortex, anterior insula, basal ganglia, and lateral temporal cortex) that have been commonly identified as being involved in the stop signal task using univariate analyses. In particular, the right prefrontal region that is detected by BSL but missed by the other methods has been widely noted to be involved in this task (Aron et al., 2004).

7 Conclusion

We develop a novel Bayesian model for structured predictive modeling of functional neuroimaging data, designed to jointly capture the block spatial sparsity and spatial smoothness properties of the neural signal. We also propose an efficient model representation for the small sample high dimensional domain and develop efficient inference, parameter estimation and prediction procedures. BSL is applied to simulated data and real fMRI data, and it is shown to outperform alternative models that focus on spatial sparsity alone.

Acknowledgments

We thank Gael Varoquaux for helpful discussions and Python code for randomized ward lasso.

References

N.M. Alpert, D. Berdichevsky, Z. Levin, E.D. Morris, and A.J. Fischman. Improved methods for image registration. *NeuroImage*, 3(1):10 – 18, 1996. ISSN 1053-8119.

A. R. Aron and R. A. Poldrack. Cortical and subcortical contributions to Stop signal response inhibition: role of the subthalamic nucleus. *J. Neurosci.*, 26(9):2424–2433, Mar 2006.

A. R. Aron, T. W. Robbins, and R. A. Poldrack. Inhibition and the right inferior frontal cortex. *Trends in cognitive sciences*, 8(4):170–177, 2004.

Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 0387310738.

Melissa K Carroll, Guillermo A Cecchi, Irina Rish, Rahul Garg, and A Ravishankar Rao. Prediction and interpretation of distributed neural activity with sparse models. *NeuroImage*, 44(1):112–122, 2009.

G. Casella. An introduction to empirical bayes data analysis. *American Statistician*, pages 83–87, 1985.

Savoy RL. Cox, D. Functional magnetic resonance imaging (fMRI) brain reading: detecting and classifying distributed patterns of fMRI activity in human visual cortex. *NeuroImage*, 19(2):261–270, June 2003.

S.J. Hanson, T. Matsuka, and J.V. Haxby. Combinatorial codes in ventral temporal lobe for object recognition: Haxby (2001) revisited: is there a. *Neuroimage*, 23(1): 156–166, 2004.

Haitham Hassanieh, Piotr Indyk, Dina Katabi, and Eric Price. Simple and practical algorithm for sparse fourier transform. In *SODA*, pages 1183–1194, 2012.

Nicolai Meinshausen and Peter Bhlmann. Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(4):417–473, 2010. ISSN 1467-9868.

Tom M. Mitchell, Rebecca Hutchinson, Radu S. Niculescu, Francisco Pereira, Xuerui Wang, Marcel Just, and Sharlene Newman. Learning to decode cognitive states from brain images. *Mach. Learn.*, 57(1-2):145–175, October 2004.

C.N. Morris. Parametric empirical bayes inference: theory and applications. *Journal of the American Statistical Association*, pages 47–55, 1983.

Kenneth A. Norman, Sean M. Polyn, Greg J. Detre, and James V. Haxby. Beyond mind-reading: multi-voxel pattern analysis of fMRI data. *Trends in Cognitive Sciences*, 10(9):424–430, September 2006.

A.V. Oppenheim and R.W. Schafer. *Discrete-time signal processing*. Prentice-Hall signal processing series. Prentice Hall, 1989. ISBN 9780132162920.

Mijung Park and Jonathan W. Pillow. Receptive field inference with localized priors. *PLoS Comput Biol*, 7(10): e1002219, 10 2011.

Mijung Park, Greg Horwitz, and Jonathan W. Pillow. Active learning of neural response functions with gaussian processes. In J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 2043–2051. NIPS, 2011.

Francisco Pereira, Tom Mitchell, and Matthew Botvinick. Machine learning classifiers and fMRI: A tutorial overview. *NeuroImage*, 45(1, Supplement 1):S199–S209, 2009. Mathematics in Brain Imaging.

Russell A Poldrack, Yaroslav O Halchenko, and Stephen José Hanson. Decoding the large-scale structure of brain function by classifying mental states across individuals. *Psychological Science*, 20(11):1364–1372, 2009.

Russell J.A. Poldrack. Inferring mental states from neuroimaging data: From reverse inference to large-scale decoding. *Neuron*, 72(5):692–697, 2011.

Carl E. Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning series)*. The MIT Press, November 2005. ISBN 026218253X.

Maneesh Sahani and Jennifer F. Linden. Evidence optimization techniques for estimating stimulus-response functions. In *NIPS*, pages 301–308, 2002.

Michael E. Tipping. Sparse bayesian learning and the relevance vector machine. *J. Mach. Learn. Res.*, 1:211–244, September 2001.

Gaël Varoquaux, Alexandre Gramfort, and Bertrand Thirion. Small-sample brain mapping: sparse recovery on spatially correlated designs with randomization and clustering. In Langford John and Pineau Joelle, editors, *International Conference on Machine Learning*. Andrew McCallum, June 2012.

Martin J. Wainwright. Sharp thresholds for high-dimensional and noisy sparsity recovery using l1-constrained quadratic programming (lasso). *IEEE Trans. Inf. Theor.*, 55(5):2183–2202, May 2009. ISSN 0018-9448. doi: 10.1109/TIT.2009.2016018.

David Wipf and Srikantan Nagarajan. A new view of automatic relevance determination. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1625–1632. MIT Press, Cambridge, MA, 2008.

Peng Zhao and Bin Yu. On model selection consistency of lasso. *J. Mach. Learn. Res.*, 7:2541–2563, December 2006. ISSN 1532-4435.

Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B*, 67:301–320, 2005.