Journal of Machine Learning Research 20 (2019) 1-43 Submitted 12/17; Revised 1/19; Published 5/19

Dependent relevance determination for smooth and structured sparse regression

Anqi Wu ANQIW@PRINCETON.EDU
Princeton Neuroscience Institute
Princeton University
Princeton, NJ 08544, USA

Oluwasanmi Koyejo SANMI@ILLINOIS.EDU
Beckman Institute for Advanced Science and Technology
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, Illinois, 61801, USA

Jonathan Pillow PILLOW@PRINCETON.EDU
Princeton Neuroscience Institute
Princeton University
Princeton, NJ 08544, USA

Editor: David Wipf

Abstract

In many problem settings, parameter vectors are not merely sparse but dependent in such a way that non-zero coefficients tend to cluster together. We refer to this form of dependency as "region sparsity." Classical sparse regression methods, such as the lasso and automatic relevance determination (ARD), which model parameters as independent *a priori*, and therefore do not exploit such dependencies. Here we introduce a hierarchical model for smooth, region-sparse weight vectors and tensors in a linear regression setting. Our approach represents a hierarchical extension of the relevance determination framework, where we add a transformed Gaussian process to model the dependencies between the prior variances of regression weights. We combine this with a structured model of the prior variances of Fourier coefficients, which eliminates unnecessary high frequencies. The resulting prior encourages weights to be region-sparse in two different bases simultaneously. We develop Laplace approximation and Monte Carlo Markov Chain (MCMC) sampling to provide efficient inference for the posterior. Furthermore, a two-stage convex relaxation of the Laplace approximation approach is also provided to relax the inevitable non-convexity during the optimization. We finally show substantial improvements over comparable methods for both simulated and real datasets from brain imaging.

Keywords: Bayesian nonparametric, Sparsity, Structure learning, Gaussian Process, fMRI

1. Introduction

Recent work in statistics has focused on high-dimensional inference problems in which the number of parameters equals or exceeds the number of samples. We focus specifically on the linear regression setting: consider a scalar response $y_i \in \mathbb{R}$ generated from an input vector $\mathbf{x}_i \in \mathbb{R}^p$ via the linear model:

$$y_i = \mathbf{x}_i^\top \mathbf{w} + \epsilon_i, \quad \text{for} \quad i = 1, 2, \cdots, n, \tag{1}$$

with observation noise $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$. The regression (linear weight) vector $\mathbf{w} \in \mathbb{R}^p$ is the quantity of interest. This general problem is ill-posed when $n \leq p$. However, it is surprisingly tractable when $\mathbf{w}$ has special structure, such as sparsity in an appropriate basis. A large literature has provided theoretical guarantees about the solvability of such problems, as well as a suite of practical methods for solving them.

Methods based on simple sparsity such as the lasso (Tibshirani, 1996) typically treat regression weights as independent *a priori*. This neglects a statistical feature of many real-world problems, which is that non-zero weights tend to arise in local groups or clusters. In many problem settings, weights have an explicit geometric relationship, such as indexing in time (e.g., time series regression) or space (e.g., brain imaging data). If a single regression weight is non-zero, nearby weights in time or space are also likely to be non-zero. Conversely, in a region where most weights are zero, any particular coefficient is also likely to be zero. Thus, nearby weights exhibit dependencies that are not captured by independent priors. We refer to this form of dependency as *region sparsity*.

A variety of methods have been developed to incorporate local dependencies between regression weights, such as the group lasso (Yuan and Lin, 2006). However, these methods typically require the user to pre-specify the group size or to partition the weights into groups *a priori*. Such information is unavailable in many applications of interest, and hard partitioning into groups breaks dependencies between nearby coefficients that are assigned to different groups.

In this paper, we take a Bayesian approach to inferring regression weights with region-sparse structure. We introduce a hierarchical prior over $\mathbf{w}$ of the form:

$$\mathbf{u} \sim \mathcal{GP} \tag{2}$$

$$\mathbf{w}|\mathbf{u} \sim \mathcal{N}(0, C(\mathbf{u})), \tag{3}$$

where $\mathbf{u}$ is a latent vector that captures dependencies in the sparsity pattern of $\mathbf{w}$, and $\mathbf{w}|\mathbf{u}$ has a zero-mean Gaussian distribution with a diagonal covariance matrix $C(\mathbf{u})$, given by a deterministic function of $\mathbf{u}$. We use a Gaussian process (GP) prior over $\mathbf{u}$ to encode structural assumptions about region sparsity (e.g., the typical size of clusters of non-zero weights and the spacing between them). This model can be seen as an extension of automatic relevance determination (ARD), in which the elements of $\mathbf{u}$ are *a priori* independent (MacKay, 1992; Neal, 1995). We therefore refer to it as *dependent relevance determination* (DRD).

Note that region-sparsity refers only to the sparsity pattern of regression weights, i.e., the locations where they are non-zero, not to the particular values of the weights themselves.

This is reflected in the fact that we define the DRD prior covariance matrix $C(\mathbf{u})$ to be diagonal, making the weights conditionally independent given the pattern defined by $\mathbf{u}$. In many cases, however, we expect weights to be smooth as well as sparse due to the continuity of the input regressors in space or time. Most of the real datasets do exhibit spatial and temporal correlations. Coefficients usually possess contiguous regions and smoothness. Hence, we are aiming at developing a universal approach easily integrating both structured sparsity and smoothness concurrently. To incorporate smoothness, we combine the standard DRD prior with a squared exponential covariance function. The resulting prior has a non-diagonal covariance matrix that encourages smoothness as well as sparsity. We refer to this extension as *smooth dependent relevance determination* (smooth-DRD). Samples from the smooth-DRD prior have local islands of smooth and non-zero weights, surrounded by large regions of zeros. We will show that combining region-sparsity and smoothness together will significantly enhance the performance in a non-trivial way.

Unfortunately, exact inference under DRD and smooth-DRD priors is analytically intractable. We therefore introduce an approximate inference method based on a Laplace approximation to the posterior over $\mathbf{u}$, and a sampling-based inference method using Monte Carlo Markov Chain (MCMC) sampling. We also derive a two-stage convex relaxation of the Laplace approximation approach in order to overcome the effects of bad local optima.

We show experimental evaluations on 1D simulated datasets comparing the performance among different methods. In addition, the phase transition curve analyses are carried out against lasso to show the superiority of DRD and smooth-DRD in support recovery for group structure sparsity with or without smoothness. Furthermore, the DRD based priors are exploited for three brain imaging datasets. Domain expertise and current evidence in brain imaging suggest that discrimination performance is primarily driven by spatially smooth activation within spatially sparse regions, and several estimation algorithms have been proposed that exploit this structure (Grosenick et al., 2011; Michel et al., 2011; Baldassarre et al., 2012; Gramfort et al., 2013). We provide experimental comparisons to these methods, showing the superiority of DRD in practice. In particular, DRD provides spatial decoding weights for brain imaging data that are both more interpretable and achieve higher decoding performance.

Here we highlight our key contributions as follows:

- We introduce a new hierarchical model for smooth, region-sparse weight tensors. The model uses a Gaussian process to introduce dependencies between prior variances of regression weights governing localized sparsity in weights and simultaneously imposes smoothness by integrating a smoothness-inducing covariance function into the prior distribution of weights.

- We describe two methods for inferring the model parameters: one based on the Laplace approximation and a second based on MCMC. We propose a fast approximate inference method based on the Laplace approximation involving a novel two-stage convex relaxation of the log posterior in order to overcome the effects of bad local optima.

- We show phase transition curves governing the transition from imperfect to near-perfect recovery for lasso and DRD estimators, revealing that group structure and smoothness can have a major impact on the recoverability of sparse signals.

This paper is organized as follows. In Sec. 2, we review the related structured sparsity literature. In Sec. 3, we introduce our new region-sparsity and smoothness inducing priors. In Sec. 4, we propose two approaches to Bayesian inference for parameter estimation, the evidence optimization via Laplace approximation and the MCMC sampling. A two-stage convex relaxation of the Laplace approximation approach is also introduced to alleviate the non-convexity with a more robust two-stage convexity. Sec. 5 introduces a detailed analysis of the structured sparsity and smoothness properties of the DRD based priors and the other methods that can be used for this purpose. Sec. 6 presents the phase transition analysis for lasso and DRD estimators. Sec. 7 shows some experiments on three real brain imaging datasets, comparing different methods that can be used for structured sparsity. Finally, Sec. 8 presents the conclusion and discussion of this work.

2. Related work

The classic method for sparse variable selection is the lasso, introduced by Tibshirani (1996), which places an l_1 penalty on the regression weights. This method can be interpreted as a *maximum a posteriori* (MAP) estimate under a Laplace (or double-exponential) prior. A fully Bayesian treatment of this model was later developed by Park and Casella (2008). A variety of Bayesian methods based on other sparsity-inducing prior distributions have been developed, including the horseshoe prior (Carvalho et al., 2009), which uses a continuous density with an infinitely tall spike at the origin and heavy tails, and the spike-and-slab prior (Mitchell and Beauchamp, 1988) which consists of a weighted mixture of a delta function (the spike) and a broad Gaussian (the slab), both centered at the origin.

Another approach to sparse variable selection comes from empirical Bayes (also known as evidence optimization or “type-II” marginal likelihood). These methods rely on a two-step inference procedure: (1) optimize hyperparameters governing the sparsity pattern via ascent of the marginal likelihood; and then (2) compute MAP estimates of the parameters given the hyperparameters. The most popular such estimator is automatic relevance determination (ARD), which prunes unnecessary coefficients by optimizing the precision of each regression coefficient under a Gaussian model (MacKay, 1992; Neal, 1995). The relevance vector machine (RVM) was later formulated as a general Bayesian framework for obtaining sparse solutions to regression and classification tasks (Tipping, 2001). The RVM has an identical functional form to the support vector machine, but provides probabilistic analysis. Tipping and Faul (2003) then was proposed for RVM to scale up its training procedure.

All these methods can be interpreted as imposing a sparse and independent prior on the regression weights. The resulting posterior over weights has high concentration near the axes, so that many weights end up at zero unless forced away strongly by the likelihood.

In the field of structured sparsity learning, group lasso is the most straightforward extension of lasso to capture sparsity existing across collections of variables (Yuan and Lin, 2006).

They achieved the group sparse structure by introducing an l_1 penalty on the l_2 norms of each group. Moreover, Huang et al. (2011) generalized the group sparsity idea by using coding complexity regularization methods associated with the structure. A variety of other papers have proposed alternative approaches to correlated or structured regularization (Jacob et al., 2009; Liu et al., 2009; Kim and Xing, 2009; Friedman et al., 2010; Jenatton et al., 2011; Kowalski et al., 2013).

Previous literature has also explored Bayesian methods for structured sparse inference. A common strategy is to introduce a latent multivariate Gaussian that controls the correlation structure governing conditionally independent densities over coefficients. Gerven et al. (2009) extended the univariate Laplace prior to a novel multivariate Laplace distribution represented as a scale mixture that induces coupling. Hernández-Lobato and Hernández-Lobato (2013) described a similar approach that results in a marginally horseshoe prior. Several other papers have proposed dependent generalizations of the spike-and-slab prior. Hernández-Lobato et al. (2013) described a group spike-and-slab distribution using a multivariate Bernoulli distribution over the indicators of the spikes associated with a group specification. Subsequently, Andersen et al. (2014, 2015) relaxed the hard-coded group specification by encoding the structure with a generic covariance function. Meanwhile, Engelhardt and Adams (2014) introduced a Bayesian model for structured sparsity that uses a Gaussian process (GP) to control the mixing weights of the spike and slab prior in proportion to feature similarity. Apart from imposing the correlation structure on the independent spike and slab elements, Yu et al. (2012) put forward a hierarchical Bayesian framework with the mixing weights of the cluster patterns generated from Beta distributions. Our work is most similar to Engelhardt and Adams (2014) and Andersen et al. (2015), except that we use an ARD-like approach with a conditionally Gaussian density over coefficients instead of a spike and slab prior. Our work is also the first that we are aware of that simultaneously captures sparsity and smoothness.

3. Dependent relevance determination (DRD) priors

In this section, we introduce the DRD prior and the smooth-DRD prior, an extension to incorporate smoothness of regression weights. We focus on the linear regression setting with conditional responses distributed as:

$$\mathbf{y}|X, \mathbf{w}, \sigma^2 \sim \mathcal{N}(\mathbf{y}|X\mathbf{w}, \sigma^2 I), \tag{4}$$

where $X = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^\top \in \mathbb{R}^{n \times p}$ denotes the design matrix, $\mathbf{y} = [y_1, \cdots, y_n]^\top \in \mathbb{R}^n$ is the observation vector, and σ^2 is the observation noise variance, where p is the dimension of the input vectors and n is the number of samples.

3.1. Automatic relevance determination

The relevance determination framework includes a family of estimators that rely on a zero-mean multivariate normal prior:

$$\mathbf{w}|\boldsymbol{\theta} \sim \mathcal{N}(0, C(\boldsymbol{\theta})), \tag{5}$$

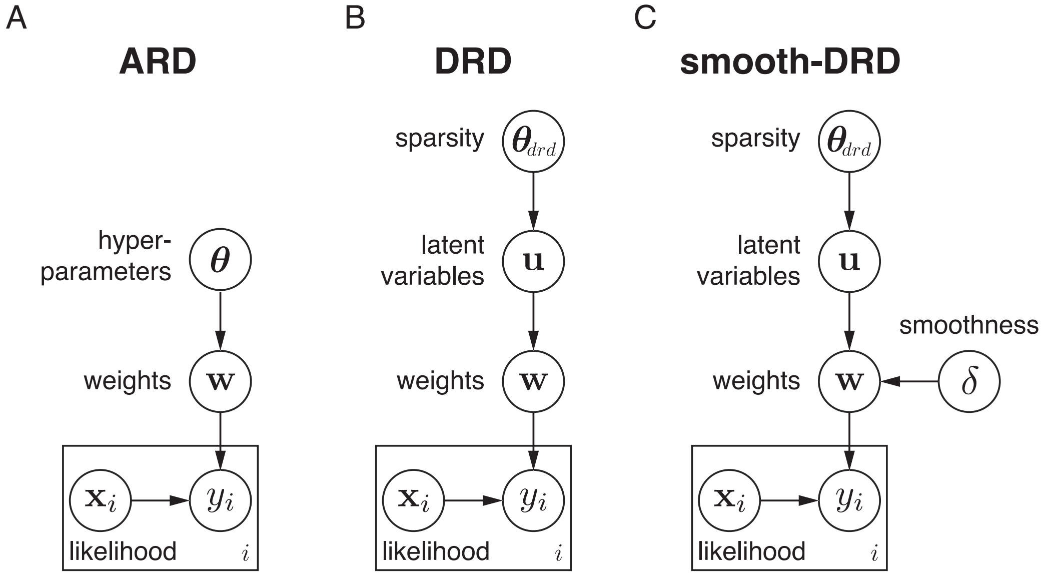

Figure 1: Graphical models for ARD, DRD and smooth-DRD.

where the prior covariance matrix $C(\boldsymbol{\theta})$ is a function of some hyperparameters $\boldsymbol{\theta}$. The form of the dependence of C on θ leads to different forms of assumed structure, including sparsity (Tipping, 2001; Faul and Tipping, 2002; Wipf and Nagarajan, 2008), smoothness (Sahani and Linden, 2003; Schmolck, 2008), or locality (Park and Pillow, 2011).

Automatic relevance determination (ARD) defines the prior covariance to be diagonal, $C_{ii} = \theta_i^{-1}$, where a distinct hyperparameter θ_i specifies the prior precision for the i'th regression coefficient. ARD places an independent improper gamma prior on each hyperparameter, $\theta_i \sim \text{gamma}(0, 0)$, and performs inference for $\{\theta_i\}$ by maximizing the marginal likelihood. This sends many θ_i to infinity, pruning the corresponding coefficients out of the model. A typical graphical model for ARD is presented in Fig. 1A. The independence assumption in the prior over hyperparameters means that there is no tendency for nearby coefficients to remain in or be pruned from the model. This is the primary shortcoming that our method seeks to overcome.

3.2. DRD: A hierarchical extension of ARD

We extend the standard ARD model by adding a level of hierarchy. Instead of directly optimizing hyperparameters that control sparsity of each weight, as in ARD, we introduce a latent vector governed by a GP prior to capture dependencies in the sparsity pattern over weights (see Fig. 1B). Let $\mathbf{u} \in \mathbb{R}^p$ denote a latent vector distributed according to a GP prior

$$\mathbf{u} \sim \mathcal{GP}(b\mathbf{1}, K), \tag{6}$$

where $b \in \mathbb{R}$ is the scalar mean, $\mathbf{1}$ is a length-p vector of ones, and covariance matrix K is determined by a squared exponential kernel. The i, j'th entry of K is given by

$$K_{ij} = \rho \exp\left(-\frac{||\chi_i - \chi_j||^2}{2l^2}\right), \tag{7}$$

where χ_i and χ_j are the spatial locations of weights w_i and w_j, respectively, and kernel hyperparameters are the marginal variance $\rho > 0$ and length scale $l > 0$. Samples from this GP on a grid of locations $\{\chi_i\}$ are smooth on the scale of l, and have mean b and marginal variance ρ.

To obtain a prior over region-sparse weight vectors, we transform $\mathbf{u}$ to the positive values via a nonlinear function f, and the transformed latent vector $\mathbf{g} = f(\mathbf{u})$ forms the diagonal of a diagonal covariance matrix for a zero-mean Gaussian prior over the weights:

$$C_{drd} = \text{diag}\Big[f(\mathbf{u})\Big], \tag{8}$$

where f is a monotonically increasing function that squashes negative values of $\mathbf{u}$ to near zero. Here we will mainly consider the exponential function $f(u) = \exp(u)$, but we will also consider "soft-rectification" function $f(u) = \log(1+\exp(u))$ in the experiment for numerical stability. When the GP mean b is very negative relative to the prior standard deviation $\sqrt{\rho}$, most elements of $\mathbf{g}$ will be close to zero, resulting in weights $\mathbf{w}$ with a high degree of sparsity (i.e., few weights far from zero). The length scale l determines the smoothness of samples $\mathbf{u}$ and thereby determines the typical width of bumps in the prior variance $\mathbf{g}$. We denote the set of hyperparameters governing the GP prior on $\mathbf{u}$ by $\boldsymbol{\theta}_{drd} = \{b, \rho, l\}$. Fig. 2A shows a depiction of sampling from the DRD generative model.

3.3. Smooth-DRD

The standard DRD model imposes smooth dependencies in the prior variances of the regression weights, but the weights themselves remain uncorrelated (as reflected by the fact that the covariance C_{drd} is diagonal). In many settings, however, we expect weights to exhibit smoothness in addition to region sparsity. To capture this property, we can augment DRD with a second Gaussian process, denoted as smooth-DRD, that induces smoothness, contributing off-diagonal structure to the prior covariance matrix while preserving the marginal variance pattern imposed by DRD (see Fig. 1C).

Let Σ denote a covariance matrix governed by a standard squared-exponential GP kernel:

$$\Sigma_{ij} = \exp\left(-\frac{||\chi_i - \chi_j||^2}{2\delta^2}\right), \tag{9}$$

with length scale δ and marginal variance set to 1. Then we define the *smooth-DRD* covariance as the "sandwich" matrix given by:

$$C_{smooth-DRD} = C_{drd}^{\frac{1}{2}} \, \Sigma \, C_{drd}^{\frac{1}{2}}, \tag{10}$$

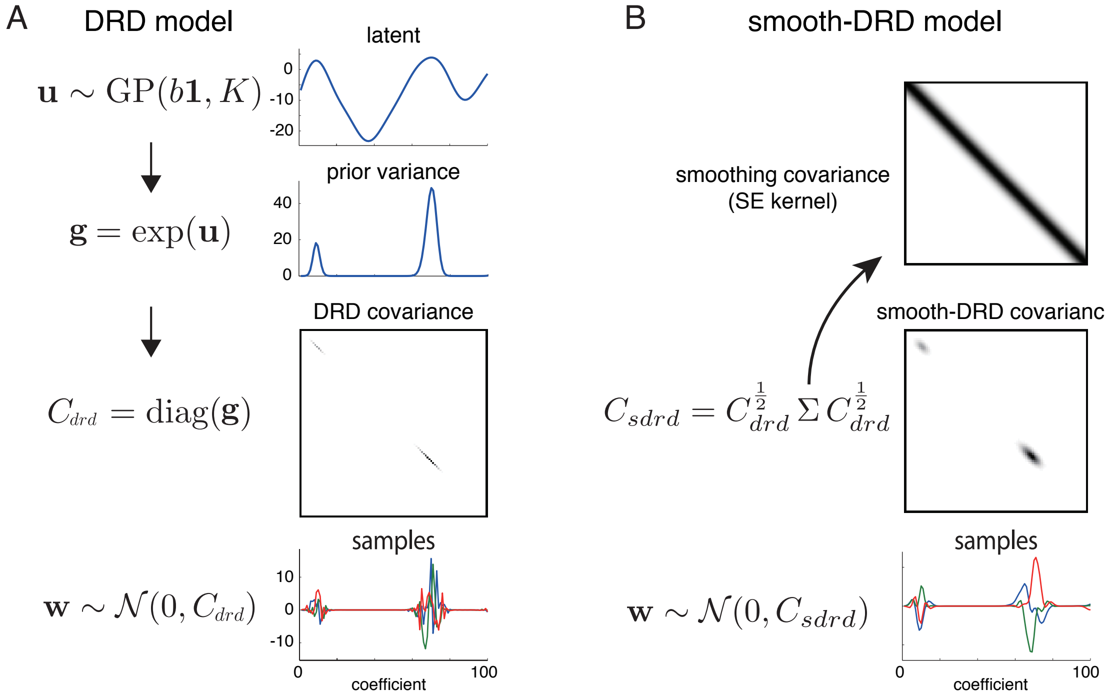
A
DRD model
$\mathbf{u} \sim \mathrm{GP}(b\mathbf{1}, K)$
latent
0
-10
-20
$\mathbf{g} = \exp(\mathbf{u})$
prior variance
40
20
0
$C_{drd} = \mathrm{diag}(\mathbf{g})$
DRD covariance
$\mathbf{w} \sim \mathcal{N}(0, C_{drd})$
samples
10
0
-10
0
100
coefficient
B
smooth-DRD model
smoothing covariance
(SE kernel)
smooth-DRD covariance
$C_{sdrd} = C_{drd}^{\frac{1}{2}} \Sigma C_{drd}^{\frac{1}{2}}$
$\mathbf{w} \sim \mathcal{N}(0, C_{sdrd})$
samples
0
100
coefficient

Figure 2: The sampling procedures for the generative models of DRD and smooth-DRD.

where $C_{drd}^{\frac{1}{2}}$ is simply the matrix square root of the diagonal covariance matrix C_{drd}. The resulting matrix has the same diagonal entries as C_{drd}, but has off-diagonal structure governed by Σ that induces smoothness. This matrix is positive semi-definite because, for all $\mathbf{x} \in \mathbb{R}^p$, $\mathbf{x}^\top C_{smooth-DRD}\mathbf{x} = (C_{drd}^{\frac{1}{2}}\mathbf{x})^\top \Sigma (C_{drd}^{\frac{1}{2}}\mathbf{x}) \geq 0$, due to the positive semi-definiteness of Σ. It is therefore a valid covariance matrix. Fig. 2B shows a depiction of sampling from the smooth-DRD generative model. In the following, we will let $\boldsymbol{\theta}$ denote the entire hyperparameter set for the smooth-DRD prior and the noise variance, where $\boldsymbol{\theta} = \{\boldsymbol{\theta}_{drd}, \delta, \sigma^2\}$.

4. Parameter estimation

In this section, we describe two methods for inference under the DRD and smooth-DRD priors: (1) empirical Bayesian inference via evidence optimization using the Laplace approximation; and (2) fully Bayesian inference via MCMC sampling. The first seeks to find the MAP estimate of the latent vector $\mathbf{u}$ governing region sparsity via optimization of the log marginal likelihood, and then provides a conditional MAP estimate of the weights $\mathbf{w}$. The second uses MCMC sampling to integrate over $\mathbf{u}$ and provides the posterior mean of $\mathbf{w}$ given the data via an average over samples.

4.1. Empirical Bayes inference with Laplace approximation

The likelihood $p(\mathbf{y}|X, \mathbf{w}, \sigma^2)$ (eq. 4) and the prior $p(\mathbf{w}|\mathbf{u}, \boldsymbol{\theta}_{drd}, \delta)$ (eq. 5) are both Gaussian given the latent variables $\mathbf{u}$ and hyperparameters $\boldsymbol{\theta}$, giving a conditionally Gaussian posterior over the regression weights:

$$p(\mathbf{w}|X, \mathbf{y}, \mathbf{u}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{w}}, \Lambda_{\mathbf{w}}), \tag{11}$$

with covariance and mean given by

$$\Lambda_{\mathbf{w}} = (\tfrac{1}{\sigma^2} X^\top X + C^{-1})^{-1}, \quad \boldsymbol{\mu}_{\mathbf{w}} = \tfrac{1}{\sigma^2} \Lambda_{\mathbf{w}} X^\top \mathbf{y}, \tag{12}$$

where prior covariance matrix C is a function of $\mathbf{u}$ and $\boldsymbol{\theta}$. The posterior mean $\boldsymbol{\mu}_{\mathbf{w}}$ is also the MAP estimate of $\mathbf{w}$ given latent vector $\mathbf{u}$ and hyperparameters $\boldsymbol{\theta}$.

Empirical Bayes inference involves setting the hyperparameters by maximizing the marginal likelihood or evidence, given by

$$p(\mathbf{y}|X, \boldsymbol{\theta}) = \iint p(\mathbf{y}|X, \mathbf{w}, \sigma^2) p(\mathbf{w}|\mathbf{u}, \delta) p(\mathbf{u}|\boldsymbol{\theta}_{drd})\, d\mathbf{w}\, d\mathbf{u}. \tag{13}$$

We can take the integral over $\mathbf{w}$ analytically due to the conditionally Gaussian prior and the likelihood, giving the simplified expression

$$p(\mathbf{y}|X, \boldsymbol{\theta}) = \int p(\mathbf{y}|X, \mathbf{u}, \sigma^2, \delta)\, p(\mathbf{u}|\boldsymbol{\theta}_{drd})\, d\mathbf{u}, \tag{14}$$

where the conditional evidence given $\mathbf{u}$ is a normal density evaluated at $\mathbf{y}$,

$$p(\mathbf{y}|X, \mathbf{u}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{y}|\mathbf{0}, XCX^\top + \sigma^2 I). \tag{15}$$

However, the integral over $\mathbf{u}$ has no analytic form. We therefore resort to the Laplace's method to approximate this integral.

4.1.1. Laplace approximation

Laplace's method provides a technique for approximating intractable integrals using a second-order Taylor expansion in $\mathbf{u}$ of the log of the integrand in (eq. 14). This method is equivalent to approximating the posterior over $\mathbf{u}$ given $\boldsymbol{\theta}$ by a Gaussian centered on its mode (MacKay (2003), chap. 27). The exact posterior is given by Bayes' rule:

$$p(\mathbf{u}|X, \mathbf{y}, \boldsymbol{\theta}) = \frac{1}{Z} p(\mathbf{y}|X, \mathbf{u}, \sigma^2, \delta) p(\mathbf{u}|\boldsymbol{\theta}_{drd}), \tag{16}$$

where the normalizing constant, $Z = p(\mathbf{y}|X, \boldsymbol{\theta})$, is the marginal likelihood we wish to compute. The Gaussian approximation to the posterior is

$$p(\mathbf{u}|X, \mathbf{y}, \boldsymbol{\theta}) \approx \mathcal{N}(\mathbf{m}_{\mathbf{u}}, \Lambda_{\mathbf{u}}), \tag{17}$$

where $\mathbf{m_u}$ is the posterior mode and $\Lambda_\mathbf{u}$ is a local approximation to the posterior covariance. Substituting this approximation into (eq. 16), we can directly solve for Z:

$$Z \approx \frac{p(\mathbf{y}|X, \mathbf{u}, \sigma^2, \delta)p(\mathbf{u}|\boldsymbol{\theta}_{drd})}{\mathcal{N}(\mathbf{m_u}, \Lambda_\mathbf{u})}. \tag{18}$$

The right-hand-side of this expression can be evaluated at any $\mathbf{u}$, but it is conventional to use the mode, $\mathbf{u} = \mathbf{m_u}$, given that this is where the approximation is most accurate.

To compute the Laplace approximation, we first numerically optimize the log of the posterior (eq. 16) to find its mode:

$$\mathbf{m_u} = \arg\max_{\mathbf{u}} \left[\log p(\mathbf{y}|X, \mathbf{u}, \sigma^2, \delta) + \log p(\mathbf{u}|\boldsymbol{\theta}_{drd})\right], \tag{19}$$

where the first term is the log of the conditional evidence given $\mathbf{u}$ (eq. 15),

$$\log p(\mathbf{y}|X, \mathbf{u}, \sigma^2, \delta) = -\frac{1}{2}\log|XCX^\top + \sigma^2 I| - \frac{1}{2}\mathbf{y}^\top(XCX^\top + \sigma^2 I)^{-1}\mathbf{y} + const, \tag{20}$$

and the second is the log of the GP prior for $\mathbf{u}$,

$$\log p(\mathbf{u}|\boldsymbol{\theta}_{drd}) = -\frac{1}{2}(\mathbf{u} - b\mathbf{1})^\top K^{-1}(\mathbf{u} - b\mathbf{1}) - \frac{1}{2}\log|K| + const. \tag{21}$$

We use quasi-Newton methods to optimize this objective function because the fixed point methods developed for ARD (e.g., MacKay (1992); Tipping and Faul (2003)), which operate on one element of the prior precision vector at a time, are inefficient due to the strong dependencies induced by the GP prior. However, because this high-dimensional optimization problem is non-convex, we also formulate a novel approach for optimizing $\mathbf{u}$ using a two-stage convex relaxation inspired by Wipf and Nagarajan (2008). We will present the method in Sec. 4.1.2.

Given the mode of the log-posterior $\mathbf{m_u}$, the second step to computing the Laplace-based approximation to the marginal likelihood is to compute the Hessian (2nd derivative matrix) of the log-posterior at $\mathbf{m_u}$. The negative inverse of the Hessian gives us the posterior covariance for the Laplace approximation (eq. 17):

$$\Lambda_\mathbf{u} = \left(-\frac{\partial^2}{\partial\mathbf{u}\partial\mathbf{u}^\top}\left[\log p(\mathbf{y}|X, \mathbf{u}, \sigma^2, \delta) + \log p(\mathbf{u}|\boldsymbol{\theta}_{drd})\right]\right)^{-1}. \tag{22}$$

See Appendix A for the explicit derivation of Hessian for the DRD model.

Given these ingredients, we can now write down the approximation to the log marginal likelihood (eq. 18):

$$\log p(\mathbf{y}|X, \boldsymbol{\theta}) \approx \log p(\mathbf{y}|X, \mathbf{m_u}, \sigma^2, \delta) + \log p(\mathbf{m_u}|\boldsymbol{\theta}_{drd}) + \tfrac{1}{2}\log|\Lambda_\mathbf{u}| + const, \tag{23}$$

where the first term is simply the log conditional evidence (eq. 20) with prior covariance C evaluated at $\mathbf{m_u}$.

It is this log-marginal likelihood that we seek to optimize in order to learn hyperparameters $\boldsymbol{\theta}$. The key difficulty is that the Laplace approximation parameters $\mathbf{m_u}$ and $\Lambda_\mathbf{u}$ depend implicitly on $\boldsymbol{\theta}$ (since $\mathbf{m_u}$ is determined by numerical optimization at a fixed value of $\boldsymbol{\theta}$), making it impractical to evaluate their derivatives with respect to $\boldsymbol{\theta}$. To address this problem, we introduce a method for partially decoupling the Laplace approximation from the hyperparameters (Sec. 4.1.3).

4.1.2. A two-stage convex relaxation to Laplace Approximation

The optimization for $\mathbf{m_u}$ (eq. 19), the mode of the posterior over the latent vector $\mathbf{u}$, is a critical step for computing the Laplace approximation. However, the negative log-posterior is a non-convex function in $\mathbf{u}$, meaning that there is no guarantee of obtaining the global minimum. In this section, DRD resembles the original ARD model. Neither of the two most popular optimization methods for ARD, MacKay's fixed-point method (MacKay, 1992) and Tipping and Faul's fast-ARD (Tipping and Faul, 2003), are guaranteed to converge to a local minimum or even a fixed point of the log-posterior.

In this section, we introduce an alternative formulation of the cost function in (eq. 19) using an auxiliary function: this provides a tight convex upper bound that can be optimized more easily. The technique is similar to the iterative re-weighted l_1 formulation of ARD in Wipf and Nagarajan (2008).

Let $\mathcal{L}(\mathbf{u})$ denote the sum of terms in the negative log-posterior (eq. 19) that involve $\mathbf{u}$,

$$\mathcal{L}(\mathbf{u}) = \frac{1}{2}\log|XCX^\top + \sigma^2 I| + \frac{1}{2}\mathbf{y}^\top(XCX^\top + \sigma^2 I)^{-1}\mathbf{y} + \frac{1}{2}(\mathbf{u} - b\mathbf{1})^\top K^{-1}(\mathbf{u} - b\mathbf{1}), \quad (24)$$

where $C = \text{diag}(e^{\mathbf{u}})$. We denote the three terms it contains as:

$$\begin{aligned}
\mathcal{L}_1(\mathbf{u}) &= \frac{1}{2}\log|X\text{diag}(e^{\mathbf{u}})X^\top + \sigma^2 I| && (25)\\
\mathcal{L}_2(\mathbf{u}) &= \frac{1}{2}\mathbf{y}^\top(X\text{diag}(e^{\mathbf{u}})X^\top + \sigma^2 I)^{-1}\mathbf{y} && (26)\\
\mathcal{L}_3(\mathbf{u}) &= \frac{1}{2}(\mathbf{u} - b\mathbf{1})^\top K^{-1}(\mathbf{u} - b\mathbf{1}). && (27)
\end{aligned}$$

Here $\mathcal{L}_1(\mathbf{u})$ and $\mathcal{L}_3(\mathbf{u})$ are both convex in $\mathbf{u}$ (see proof in Appendix B). We can derive a tight convex upper bound for $\mathcal{L}_2(\mathbf{u})$, thus providing a tight convex upper bound for $\mathcal{L}(\mathbf{u})$.

We know that $\mathcal{L}_2(\mathbf{u})$ is non-convex, but we are interested in rewriting it using concave duality. Let $\mathbf{h}(\mathbf{u}) : \mathbb{R}^p \to \Omega \subset \mathbb{R}^p$ be a mapping with range Ω, which may or may not be a one-to-one map. We assume that there exists a concave function $\Phi(\boldsymbol{\eta}) : \Omega \to \mathbb{R}, \forall \boldsymbol{\eta} \in \Omega$, such that $\mathcal{L}_2(\mathbf{u}) = \Phi(\mathbf{h}(\mathbf{u}))$ holds. To exploit this technique, we first rewrite $\mathcal{L}_2$ using the matrix inverse lemma (Higham, 2002) as:

$$\mathcal{L}_2(\mathbf{u}) = \frac{1}{2\sigma^2}\mathbf{y}^\top\mathbf{y} - \frac{1}{2\sigma^4}\mathbf{y}^\top X\left(\frac{1}{\sigma^2}X^\top X + \text{diag}(e^{-\mathbf{u}})\right)^{-1}X^\top\mathbf{y}. \quad (28)$$

Then, setting $\mathbf{h}(\mathbf{u}) = e^{-\mathbf{u}}$, which is convex in $\mathbf{u}$, we have

$$\mathcal{L}_2(\mathbf{u}) = \Phi(\mathbf{h}(\mathbf{u})) = \frac{1}{2\sigma^2}\mathbf{y}^\top\mathbf{y} - \frac{1}{2\sigma^4}\mathbf{y}^\top X\left(\frac{1}{\sigma^2}X^\top X + \text{diag}(\mathbf{h}(\mathbf{u}))\right)^{-1}X^\top\mathbf{y}. \quad (29)$$

Algorithm 1 A two-stage convex relaxation method for DRD Laplace approximation

```
**Input:** $X, \mathbf{y}, \boldsymbol{\theta} = \{\sigma^2, \delta, b, \rho, l\}$
**Output:** $\hat{\mathbf{u}}$
initialize dual variable $\hat{\mathbf{z}}_i = 1, \forall i = 1, 2, ..., p$
Repeat the following two steps until convergence:
1. Fix $\hat{\mathbf{z}}$, let $\hat{\mathbf{u}} = \text{argmin}_{\mathbf{u}\in\mathbb{R}^p} \left[\mathbf{z}^\top \mathbf{h}(\mathbf{u}) + \mathcal{L}_1(\mathbf{u}) + \mathcal{L}_3(\mathbf{u})\right]$ in (eq. 33)
2. Fix $\hat{\mathbf{u}}$, let $\hat{\mathbf{z}} = \nabla_{\boldsymbol{\eta}} \Phi(\boldsymbol{\eta})|_{\boldsymbol{\eta}=\mathbf{h}(\hat{\mathbf{u}})}$ in (eq. 34)
```

This expression is concave in $\mathbf{h}(\mathbf{u})$ (inverse of a matrix is convex), and thus can be expressed as a minimum over upper-bounding hyperplanes via

$$\mathcal{L}_2(\mathbf{u}) = \Phi(\mathbf{h}(\mathbf{u})) = \text{inf}_{\mathbf{z}\in\mathbb{R}^p} \left[\mathbf{z}^\top \mathbf{h}(\mathbf{u}) - \mathcal{L}^*_{\mathbf{h}}(\mathbf{z})\right], \tag{30}$$

where $\mathcal{L}^*_{\mathbf{h}}(\mathbf{z})$ is the concave conjugate of $\Phi(\boldsymbol{\eta})$ that is defined by the duality relationship

$$\mathcal{L}^*_{\mathbf{h}}(\mathbf{z}) = \text{inf}_{\boldsymbol{\eta}\in\mathbb{R}^p} \left[\mathbf{z}^\top \boldsymbol{\eta} - \Phi(\boldsymbol{\eta})\right], \tag{31}$$

and $\mathbf{z}$ is the dual variable. Note, however, that for our purpose it is not necessary to ever explicitly compute $\mathcal{L}^*_{\mathbf{h}}(\mathbf{z})$. This leads to the following upper-bounding auxiliary cost function

$$\Phi(\mathbf{h}(\mathbf{u}), \mathbf{z}) = \mathbf{z}^\top \mathbf{h}(\mathbf{u}) - \mathcal{L}^*_{\mathbf{h}}(\mathbf{z}) \geq \Phi(\mathbf{h}(\mathbf{u})). \tag{32}$$

Thus, it naturally admits the tight convex upper bound for $\mathcal{L}(\mathbf{u})$,

$$\mathcal{L}(\mathbf{u}, \mathbf{z}) \stackrel{\Delta}{=} \mathbf{z}^\top \mathbf{h}(\mathbf{u}) - \mathcal{L}^*_{\mathbf{h}}(\mathbf{z}) + \mathcal{L}_1(\mathbf{u}) + \mathcal{L}_3(\mathbf{u}) \geq \mathcal{L}(\mathbf{u}). \tag{33}$$

Moreover, for any fixed $\boldsymbol{\eta} = \mathbf{h}(\mathbf{u})$, it's well-known that the minimum of the right hand side of (eq. 31) is achieved at

$$\hat{\mathbf{z}} = \nabla_{\boldsymbol{\eta}} \Phi(\boldsymbol{\eta})|_{\boldsymbol{\eta}=\mathbf{h}(\mathbf{u})}. \tag{34}$$

This leads to the general optimization procedure presented in Algorithm 1. By repeatedly refining the dual parameter $\mathbf{z}$, we can obtain a repeatedly improved convex relaxation, leading to a solution superior to that of the initial convex relaxation.

Now we show the analysis of global convergence. According to the Zangwill's *Global Convergence Theorem* (Zangwill, 1969), let $\mathcal{A}(\cdot) : \mathcal{U} \to \mathcal{P}(\mathcal{U})$ be a point-to-set mapping to handle the multi-global minima case, which satisfies Steps 1 and 2 of the proposed algorithm, then

Theorem 1 *From any initialization point* $\mathbf{u}^0 \in \mathbb{R}^p$*, the sequence of parameter estimates* $\{\mathbf{u}^k\}$ *generated via* $\mathbf{u}^{k+1} \in \mathcal{A}(\mathbf{u}^k)$ *is guaranteed to converge monotonically to a local minimum (or saddle point) of* $\mathcal{L}(\mathbf{u})$*.*

Proof Let $\Gamma \in \mathcal{U}$ be a solution set. In order to use the global convergence theorem, we need to show that

1) all points $\{\mathbf{u}^k\}$ are contained in a compact set $S \in \mathcal{U}$, where $\mathcal{U}$ is $\mathbb{R}^p$ in our problem;
2) there is a continuous function Z on $\mathcal{U}$ such that
(a) if $x \notin \Gamma$, then $Z(y) < Z(x)$ for all $y \in \mathcal{A}(x)$;
(b) if $x \in \Gamma$, then $Z(y) \leq Z(x)$ for all $y \in \mathcal{A}(x)$;
3) the mapping $\mathcal{A}$ is closed at points outside Γ.

First, let's define the mapping $\mathcal{A}$ to be achieved by

$$\mathbf{u}^{k+1} \in \mathcal{A}(\mathbf{u}^k) = \operatorname{argmin}_{\mathbf{u}\in\mathbb{R}^p} \mathcal{F}(\mathbf{u}, \mathbf{u}^k) = \operatorname{argmin}_{\mathbf{u}\in\mathbb{R}^p} {\mathbf{z}^k}^\top \mathbf{h}(\mathbf{u}) - \mathcal{L}^*_{\mathbf{h}}(\mathbf{z}) + \mathcal{L}_1(\mathbf{u}) + \mathcal{L}_3(\mathbf{u}), \tag{35}$$

where $\mathbf{z}^k = \nabla_{\boldsymbol{\eta}} \Phi(\boldsymbol{\eta})|_{\boldsymbol{\eta}=\mathbf{h}(\mathbf{u}^k)}$. We can prove that $\mathcal{F}$ is coercive, i.e., when $||\mathbf{u}|| \to \infty$, we have $\mathcal{F}(\mathbf{u}) \to \infty$ (proof in Appendix C). Therefore, the solution set of $\mathcal{F}(\mathbf{u})$ is bounded and nonempty. Accordingly, $\mathcal{A}(\mathbf{u})$ is nonempty. Using Proposition 7 in (Gunawardana and Byrne, 2005), we can further show that the point-to-set mapping $\mathcal{A}$ is closed at $\mathbf{u} \in \mathcal{U}$. Condition 3 is satisfied.

For each $\mathbf{u}^k$, $\mathbf{u}^{k+1}$ is the solution of $\mathcal{F}(\mathbf{u})$, and $\mathcal{A}(\mathbf{u})$ is a closed mapping; therefore each $\mathbf{u}^{k+1}$ belongs to a compact set. We know that the union of two compact sets is compact. Therefore, all points $\{\mathbf{u}^k\}$ are contained in a compact set $S \in \mathcal{U}$. Condition 1 is satisfied.

To prove condition 2, we must show that for any $\mathbf{u}^k$, $\mathcal{L}(\mathbf{u}^{k+1}) < \mathcal{L}(\mathbf{u}^k)$ for all $\mathbf{u}^{k+1} \in \mathcal{A}(\mathbf{u}^k)$ if $\mathbf{u}^k \notin \Gamma$; $\mathcal{L}(\mathbf{u}^{k+1}) \leq \mathcal{L}(\mathbf{u}^k)$ for all $\mathbf{u}^{k+1} \in \mathcal{A}(\mathbf{u}^k)$ if $\mathbf{u}^k \in \Gamma$. At any $\mathbf{u}^k$, the auxiliary cost function $\mathcal{F}(\mathbf{u})$ (eq. 35) is strictly tangent to $\mathcal{L}(\mathbf{u})$ at $\mathbf{u}^k$. Therefore, if $\mathbf{u}^k \notin \Gamma$, $\mathcal{L}(\mathbf{u}^k) = \mathcal{F}(\mathbf{u}^k) > \mathcal{F}(\mathbf{u}^{k+1}) \geq \mathcal{L}(\mathbf{u}^{k+1})$, thus $\mathcal{L}(\mathbf{u}^k) > \mathcal{L}(\mathbf{u}^{k+1})$; if $\mathbf{u}^k \in \Gamma$, $\mathcal{L}(\mathbf{u}^k) = \mathcal{F}(\mathbf{u}^k) = \mathcal{F}(\mathbf{u}^{k+1}) \geq \mathcal{L}(\mathbf{u}^{k+1})$, thus $\mathcal{L}(\mathbf{u}^k) \geq \mathcal{L}(\mathbf{u}^{k+1})$. Condition 2 is satisfied. ■

The algorithm could theoretically converge to a saddle point, but any minimal perturbation would easily lead to escape.

4.1.3. Decoupled Laplace approximation

To optimize the marginal likelihood for the DRD hyperparameters (eq. 23), we should ideally replace $\mathbf{m}_\mathbf{u}$ and $\Lambda_\mathbf{u}$ with explicit expressions in $\boldsymbol{\theta}$ in order to accurately compute derivatives with respect to $\boldsymbol{\theta}$. However, the deterministic formulation of such functions is intractable. We can nevertheless partially overcome this dependence by introducing a "decoupled" Laplace approximation that takes into account the dependence of $\Lambda_\mathbf{u}$ on the hyperparameters $\boldsymbol{\theta}_{drd}$. Wu et al. (2017) also proposed a conceptually similar decoupled Laplace approximation.

Specifically, we rewrite the inverse Laplace posterior covariance (eq. 22):

$$\Lambda_\mathbf{u} = (\Gamma + \Psi(\boldsymbol{\theta}_{drd}))^{-1} \tag{36}$$

where Γ is the negative Hessian of the log-likelihood (which is independent of $\boldsymbol{\theta}_{drd}$),

$$\Gamma = -\frac{\partial^2}{\partial \mathbf{u} \partial \mathbf{u}^\top} \log p(\mathbf{y}|X, \mathbf{u}, \sigma^2, \delta), \tag{37}$$

and $\Psi(\boldsymbol{\theta}_{drd})$ is the precision matrix of the prior distribution for $\mathbf{u}$,

$$\Psi(\boldsymbol{\theta}_{drd}) = -\frac{\partial^2}{\partial \mathbf{u} \partial \mathbf{u}^\top} \log p(\mathbf{u}|\boldsymbol{\theta}_{drd}) = K^{-1}, \tag{38}$$

Algorithm 2 Evidence optimization using decoupled Laplace approximation

```
Input: $X, \mathbf{y}$
Output: latents $\hat{\mathbf{u}}$, hyperparameters $\hat{\boldsymbol{\theta}} = \{\widehat{\sigma^2}, \hat{\delta}, \hat{b}, \hat{\rho}, \hat{l}\}$.
At iteration $t$:
1. Numerically optimize log-posterior for latents $\mathbf{m}_{\mathbf{u}}^t$ using (eq. 19) or Algorithm 1.
2. Compute $\Gamma^t$ using negative Hessian of the log conditional evidence (eq. 37).
3. Numerically optimize $p(\mathbf{y}|X, \boldsymbol{\theta}, \mathbf{m}_{\mathbf{u}}^t, \Gamma^t)$ (eq. 39) for $\boldsymbol{\theta}^t$.
Repeat step 1, 2 and 3 until $\{\mathbf{m}_{\mathbf{u}}, \Gamma\}$ and $\boldsymbol{\theta}$ converge.
```

which is the inverse of the GP prior covariance governing $\mathbf{u}$ (eq. 7). Substituting for $\Lambda_{\mathbf{u}}$ in (eq. 23), this gives:

$$\log p(\mathbf{y}|X, \boldsymbol{\theta}) \approx \log p(\mathbf{y}|X, \mathbf{m}_{\mathbf{u}}, \sigma^2, \delta) + \log p(\mathbf{m}_{\mathbf{u}}|\boldsymbol{\theta}_{drd}) - \tfrac{1}{2} \log |\Gamma + K^{-1}| + const. \tag{39}$$

This form decomposes the curvature at the posterior mode into the likelihood curvature and the prior curvature. In this way, the posterior curvature tracks the influence of the change in the prior curvature as we optimize the hyperparameters $\boldsymbol{\theta}$, while keeping the influence of the likelihood curvature fixed. This decoupling allows us to update the posterior without recomputing the Hessian. It will be accurate so long as the Hessian of the likelihood changes slowly over local regions in parameter space.

To optimize hyperparameters under the decoupled Laplace approximation, we fix $\mathbf{m}_{\mathbf{u}}$ and Γ using the current mode of the posterior, and optimize (eq. 39) directly for $\boldsymbol{\theta}$, incorporating the dependence of K on $\boldsymbol{\theta}_{drd}$. With this approach, the first term, $\log p(\mathbf{y}|X, \mathbf{m}_{\mathbf{u}}, \sigma^2, \delta)$, captures the dependence on σ^2 and δ; the second term, $\log p(\mathbf{m}_{\mathbf{u}}|\boldsymbol{\theta}_{drd})$, restricts $\boldsymbol{\theta}_{drd}$ around the current mode; and the third term $-\frac{1}{2} \log |\Gamma + K^{-1}|$ pushes $\boldsymbol{\theta}_{drd}$ along the second order curvature given the GP kernel. This decoupling weakens the strong dependency between $\boldsymbol{\theta}_{drd}$ and $\mathbf{m}_{\mathbf{u}}$, maintaining the accuracy of the Laplace approximation as we adjust $\boldsymbol{\theta}_{drd}$.

To ensure the accuracy of the Laplace approximation, in each iteration t, we optimize eq. (39) over a restricted region of the hyperparameter space around the previous hyperparameter setting θ^{t-1}, which allows varying within 20% of its current value on each iteration in our experiments. This prevents $\boldsymbol{\theta}$ from moving too far from the region where the current Laplace approximation ($\mathbf{m}_{\mathbf{u}}$ and Γ) is accurate. Then, based on a new hyperparameter setting $\boldsymbol{\theta}^t$, we update the Laplace approximation parameters $\mathbf{m}_{\mathbf{u}}$ and Γ. This procedure is summarized in Algorithm 2. The algorithm stops when $\{\mathbf{m}_{\mathbf{u}}, \Gamma\}$ and $\boldsymbol{\theta}$ converge. The empirical Bayes estimate is then given by the MAP estimate of the weights $\mathbf{w}_{map} = \boldsymbol{\mu}_{\mathbf{w}}$ (eq. 12) conditioned on the optimal latents $\hat{\mathbf{u}} = \mathbf{m}_{\mathbf{u}}$ and hyperparameters $\hat{\boldsymbol{\theta}}$.

4.2. Fully Bayesian inference with MCMC

An alternate approach to the empirical Bayesian inference procedure described above is to perform fully Bayesian inference using Markov Chain Monte Carlo (MCMC). Using sampling, we can compute the integrals over $\mathbf{u}$ and $\boldsymbol{\theta}$ in order to compute the posterior mean (Bayes' least squares estimates) for $\mathbf{w}$. The full posterior distribution over $\mathbf{w}$ can be written as

$$p(\mathbf{w}|X,\mathbf{y}) = \iint p(\mathbf{w}|X,\mathbf{y},\mathbf{u},\boldsymbol{\theta})p(\mathbf{u},\boldsymbol{\theta}|X,\mathbf{y})\,d\mathbf{u}\,d\boldsymbol{\theta} \tag{40}$$

$$= \iint \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}_{\mathbf{w}},\Lambda_{\mathbf{w}})p(\mathbf{u},\boldsymbol{\theta}|X,\mathbf{y})\,d\mathbf{u}\,d\boldsymbol{\theta}, \tag{41}$$

where mean $\boldsymbol{\mu}_{\mathbf{w}}$ and covariance $\Lambda_{\mathbf{w}}$ are functions of $\mathbf{u}$ and $\boldsymbol{\theta}$ (eq. 12). This suggests a Monte Carlo representation of the posterior as

$$p(\mathbf{w}|X,\mathbf{y}) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{N}\left(\mathbf{w}\;\middle|\;\boldsymbol{\mu}_{\mathbf{w}}(\mathbf{u}^{(i)},\boldsymbol{\theta}^{(i)}),\Lambda_{\mathbf{w}}(\mathbf{u}^{(i)},\boldsymbol{\theta}^{(i)})\right) \tag{42}$$

$$\mathbf{u}^{(i)},\boldsymbol{\theta}^{(i)} \sim p(\mathbf{u},\boldsymbol{\theta}|X,\mathbf{y}), \tag{43}$$

where i is the index of the samples and N is the total number of samples. We can use Gibbs sampling to alternately sample $\mathbf{u}$ and $\boldsymbol{\theta}$ from their conditional distributions given the other. The joint posterior distribution of $\mathbf{u}$ and $\boldsymbol{\theta}$ has the following proportional relationship,

$$p(\mathbf{u},\boldsymbol{\theta}|X,\mathbf{y}) \propto p(\mathbf{y}|X,\mathbf{u},\sigma^2,\delta)p(\mathbf{u}|\boldsymbol{\theta}_{drd})\text{Prior}(\boldsymbol{\theta}), \tag{44}$$

where $p(\mathbf{y}|X,\mathbf{u},\sigma^2,\delta)$ and $p(\mathbf{u}|\boldsymbol{\theta}_{drd})$ have the likelihoods given in (eq. 20) and (eq. 21), and $\text{Prior}(\boldsymbol{\theta})$ is the prior distribution for $\boldsymbol{\theta}$.

Sampling latents $\mathbf{u}|\boldsymbol{\theta}$

The first phase of Gibbs sampling is to sample $\mathbf{u}$ from the conditional distribution of $\mathbf{u}$ given $\boldsymbol{\theta}$,

$$\mathbf{u}|\boldsymbol{\theta} \sim p(\mathbf{y}|X,\mathbf{u},\sigma^2,\delta)p(\mathbf{u}|\boldsymbol{\theta}_{drd}). \tag{45}$$

This is the product of a Gaussian process prior $p(\mathbf{u}|\boldsymbol{\theta}_{drd})$ and a likelihood function $p(\mathbf{y}|X,\mathbf{u},\boldsymbol{\theta})$ that ties the latent variables $\mathbf{u}$ to the observed data. This setting meets the requirements of elliptical slice sampling (ESS), a rejection-free MCMC (Murray et al., 2009). ESS generates random elliptical loci using the Gaussian prior and then searches along these loci to find acceptable points by evaluating the data likelihood. This method takes into account strong dependencies imposed by GP covariance on the elements of the vector $\mathbf{u}$ to facilitates faster mixing. It also requires no tuning parameters, unlike alternative samplers such as Metropolis-Hastings or Hamiltonian Monte Carlo, but performs similarly to the best possible performance of a related M-H scheme. To overcome slow mixing that can result when the prior covariance is highly elongated, we apply ESS to a whitened variable using a reparametrization trick, discussed in more details in Sec. 4.3.

Sampling hyperparameters $\boldsymbol{\theta}|\mathbf{u}$

The conditional distribution for sampling $\boldsymbol{\theta}$ given $\mathbf{u}$ is

$$\boldsymbol{\theta}|\mathbf{u} \sim p(\mathbf{y}|X,\mathbf{u},\sigma^2,\delta)p(\mathbf{u}|\boldsymbol{\theta}_{drd})\text{Prior}(\boldsymbol{\theta}), \tag{46}$$

where $\boldsymbol{\theta} = \{\sigma^2, b, \rho, l, \delta\}$ contains five individual hyperparmaters. We therefore perform slice sampling for each variable conditioned on the others. We use prior distributions of the form:

$$\log(\sigma^2) \sim \mathcal{N}(m_n,\sigma_n^2), \quad b \sim \mathcal{N}(m_b,\sigma_b^2), \quad \rho \sim \Gamma(a_\rho,b_\rho), \quad l \sim \Gamma(a_l,b_l), \quad \delta \sim \Gamma(a_\delta,b_\delta). \tag{47}$$

We put a Gaussian prior on the log of σ^2 instead of σ^2. We will provide the values for these priors in Sec. 5 on synthetic experiments. To control the number of samples, we inspect burn-in of MCMC, e.g., the training error and the change of coefficient given the averaged coefficient samples.

4.3. Whitening the GP prior using reparametrization

In both Laplace approximation and MCMC frameworks, the latent vector $\mathbf{u}$ depends on the product of the conditional evidence $p(\mathbf{y}|X, \mathbf{u}, \sigma^2, \delta)$ and the GP prior $p(\mathbf{u}|\boldsymbol{\theta}_{drd})$. The GP prior (which is the primary difference between our model and standard ARD) introduces strong dependencies between $\mathbf{u}$ and GP hyperparameters, resulting in a highly elliptical joint distribution. Such distributions are often problematic for both optimization and sampling. For example, if we are trying to perform Gibbs sampling on $\mathbf{u}$ and the GP length scale hyperparameter l, and the prior is strong relative to the evidence term, the samples $\mathbf{u}|l^{(i)}$ will have smoothness strongly determined by $l^{(i)}$, and the samples $l|\mathbf{u}^{(i)}$ will in turn be strongly determined by the smoothness of the current sample $\mathbf{u}^{(i)}$. In this case, mixing will be slow, and Gibbs sampling will take a long time to explore the full posterior over different values of l.

We can overcome this difficulty with a technique known as the "reparametrization trick," which involves reparameterizing the model so that the unknown variables are independent under the prior (Murray and Adams, 2010). If we have prior $P(\mathbf{u}) = \mathcal{N}(b\mathbf{1}, K)$, then $\mathbf{u}$ can be described equivalently by a deterministic transformation of a standard normal random variable $\mathbf{v}$:

$$\mathbf{v} \sim \mathcal{N}(0, I), \quad \mathbf{u} = L\mathbf{v} + b\mathbf{1}, \tag{48}$$

where $K = LL^\top$ is the Cholesky factorization of prior covariance K.

This reparametrization simplifies Laplace-approximation-based inference by allowing a change of variables in (eq. 19) so that we directly maximize $p(\mathbf{y}|X, \mathbf{v}, \boldsymbol{\theta})\mathcal{N}(\mathbf{v}|0, I)$ for $\mathbf{v}$. This optimization problem has better conditioning, and eliminates the computational problem of computing $\mathbf{u}^\top K^{-1}\mathbf{u}$ in the log prior, which is replaced by a simple ridge penalty of the form $\mathbf{v}^\top\mathbf{v}$.

For sampling-based inference, the reparametrization allows us to improve mixing performance because the conditionals $\mathbf{v}|\boldsymbol{\theta}$ and $\boldsymbol{\theta}|\mathbf{v}$ exhibit much weaker dependencies than $\mathbf{u}|\boldsymbol{\theta}$ and $\boldsymbol{\theta}|\mathbf{u}$. Moreover, elliptical slice sampling for $\mathbf{v}|\boldsymbol{\theta}$ is more efficient because it involves loci on a sphere instead of a highly elongated ellipsoid.

4.4. Fourier dual form

A second trick for improving the computational performance of DRD is to perform optimization of the latent variable $\mathbf{u}$ (or $\mathbf{v}$) in the Fourier domain. When the GP prior induces a high degree of smoothness in $\mathbf{u}$, the prior covariance K becomes approximately low rank, meaning that it has a small number of non-negligible eigenvalues. Because the covariance function (eq. 7) is shift-invariant, the eigenspectrum of K has a diagonal representation in the Fourier domain, a consequence of Bochner's theorem (Stein, 1999; Lázaro-Gredilla et al., 2010). We can exploit this representation to optimize $\tilde{\mathbf{u}}$ (the discrete Fourier transform of $\mathbf{u}$) while ignoring Fourier components above a certain high-frequency cutoff, where this cutoff depends on the length scale l. This results in a lower-dimensional optimization problem. Fourier-domain representation of the latent vector $\mathbf{u}$ also simplifies the application of the reparametrization trick described above because the Cholesky factor L is now a diagonal matrix that can be computed analytically from the spectral density of the squared-exponential prior.

To summarize the joint application of the reparametrization and Fourier dual tricks in our model, they can be understood as allowing us to draw samples $\mathbf{u} \sim \mathcal{N}(b\mathbf{1}, K)$ via the series of transformations:

$$\tilde{\mathbf{v}} \sim \mathcal{N}(0, I), \qquad \textit{whitened Fourier domain sample} \tag{49}$$

$$\tilde{\mathbf{u}} = L\tilde{\mathbf{v}} + \tilde{\mathbf{b}}, \qquad \textit{transformed Fourier domain sample} \tag{50}$$

$$\mathbf{u} = B\tilde{\mathbf{u}}, \qquad \textit{inverse Fourier transform} \tag{51}$$

where $\tilde{\mathbf{b}}$ is the discrete Fourier transform of $b\mathbf{1}$, a vector of zeros except for a single non-zero element carrying the DC component, and B is the truncated (tall skinny) discrete inverse Fourier transform matrix mapping the low-frequency Fourier components represented in $\tilde{\mathbf{u}}$ to the space domain.

Note that the smoothness on $\mathbf{u}$, which controls the spatial scale of dependent sparsity, is different from the smoothing prior used in smooth-DRD to induce smoothness in the coefficients $\mathbf{w}$, although both can benefit from sparse Fourier-domain representation in cases where the relevant length scale is large.

5. Synthetic experiments

5.1. Simulated example with smooth and sparse weights

To illustrate and give intuition for the performance of the DRD estimator, we performed simulated experiments with a vector of regression weights in a one-dimensional space. We sampled a $p = 4000$ dimensional weight vector $\mathbf{w}$ from the smooth-DRD prior (see Fig. 1), with hyperparameters GP mean $b = -8$, GP length scale $l = 100$, GP marginal variance $\rho = 36$, smoothness length scale $\delta = 50$, measurement noise variance $\sigma^2 = 5$. We then sampled $n = 500$ responses $\mathbf{y} = X\mathbf{w} + \epsilon$, where X is an $n \times p$ design matrix with entries drawn i.i.d. from a standard normal distribution, and noise $\epsilon \sim \mathcal{N}(0, 5I)$.

Fig. 3 shows an example weight vector drawn from this prior, along with estimates obtained from a variety of different estimators:

- lasso (Tibshirani, 1996), using Least Angle Regression (LARS) implemented by glmnet[1];

[1] https://web.stanford.edu/~hastie/glmnet/glmnet_alpha.html

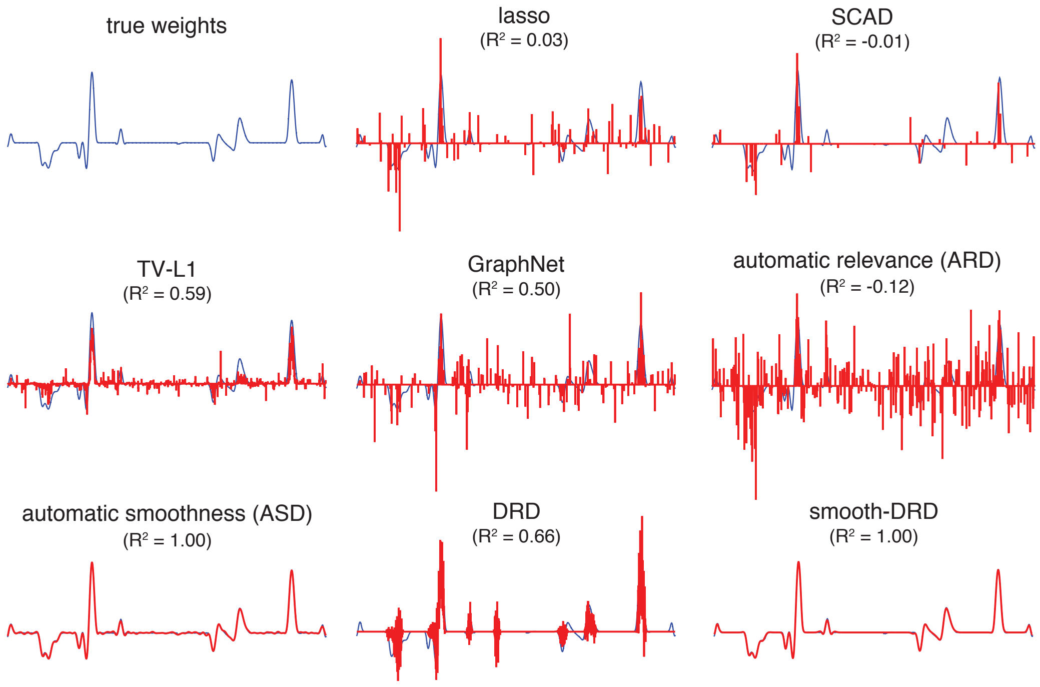

Figure 3: Example 4000-element weight vector $\mathbf{w}$ sampled from the smooth-DRD prior (upper left), and estimates obtained from different methods on a simulated dataset with $n = 500$ samples. The R^2 performance of each estimate in recovering $\mathbf{w}$ is indicated above each plot. The bottom row shows our estimators: DRD-Laplace (bottom center) and smooth-DRD-Laplace (bottom right); the other DRD and smooth-DRD estimators (not shown) achieved similar performance.

- Automatic Relevance Determination (ARD) (Neal, 1995; MacKay, 1992), implemented with the classic fixed point algorithm.

- Automatic Smoothness Determination (ASD) (Sahani and Linden, 2003), which uses numerical optimization of marginal likelihood to learn the hyperparameters of a squared exponential kernel governing $\mathbf{w}$.

- Total Variation l_1 (TV-L1) (Michel et al., 2011; Baldassarre et al., 2012; Gramfort et al., 2013), combining total variation penalty (also known as fused lasso), which imposes an l_1 penalty on the first-order differences of $\mathbf{w}$, with a standard lasso penalty.

- GraphNet (Grosenick et al., 2011), a graph-constrained elastic net, developed for spatial and temporally correlated data that yields interpretable model parameters by incorporating sparse graph priors based on model smoothness or connectivity, as well as a global sparsity inducing a prior that automatically selects important variables.

- Smoothly Clipped Absolute Deviation (SCAD) (Fan and Li, 2001), an estimator with non-convex sparsity penalty.

We computed total variation l_1 (TV-L1) and graph net (GraphNet) estimates using the Nilearn[2] package (Abraham et al., 2014). SCAD was implemented by SparseReg[3] (Zhou and Gaines, 2017). For lasso, GraphNet, TV-L1 and SCAD, we used cross-validation to set hyperparameters, whereas ARD and ASD used evidence optimization to automatically set hyperparameters. For the DRD estimators, we used evidence optimization to set hyperparameters for Laplace-approximation based estimates and used sampling to integrate over hyperparameters for MCMC-based estimates.

For the basic DRD model, which incorporates structured sparsity but not smoothness, we compared three different inference methods: (1) Laplace approximation based inference ("DRD-Laplace"); (2) Markov Chain Monte Carlo ("DRD-MCMC"); and (3) Convex relaxation based optimization ("DRD-Convex"). Lastly, for the smooth-DRD model, we used two inference methods: (4) Laplace approximation ("smooth-DRD-Laplace"); and (5) MCMC ("smooth-DRD-MCMC"). For the non-MCMC estimators, we initialized the vector of Fourier domain coefficients $\tilde{\mathbf{v}}$ (eq. 49) to values of 10^{-3} in the first iteration when learning $\mathbf{u}$. The hyper-hyperparameters in the MCMC methods (eq. 47) were set to: $m_n = -2, \sigma_n^2 = 5, m_b = -10, \sigma_b^2 = 8, a_\rho = 4, b_\rho = 5, a_l = 4, b_l = 25, a_\delta = 4, b_\delta = 25$.

Fig. 3 shows the reconstruction performance (R^2) of the true regression weight $\mathbf{w}$ for different estimators. The reconstruction performance metric for an estimate $\hat{\mathbf{w}}$ is given by $R^2 = 1 - \frac{||\mathbf{w}-\hat{\mathbf{w}}||_2^2}{||\mathbf{w}-\bar{\mathbf{w}}||_2^2}$, where $||\cdot||_2$ denotes the l_2-norm and $\bar{\mathbf{w}} = \frac{1}{p}\sum_{i=1}^{p} \mathbf{w}_i$ is the mean of vector $\mathbf{w}$. The true weight vector was sampled from the smooth DRD model. The smooth-DRD estimate achieved the best performance in terms of R^2. The ASD estimate also performed well, although the estimate was not sparse, exhibiting small wiggles where the coefficients should be zero. The standard DRD estimate recovered the support of $\mathbf{w}$ with high accuracy, but had larger error than smooth-DRD estimates due to the smoothness of the true $\mathbf{w}$. The other methods (lasso, ARD, TV-L1, GraphNet and SCAD) all had lower accuracy in recovering both the support and values of the regression weights.

To provide insight into the performance of ARD, DRD, and smooth-DRD, we plotted the inferred prior covariance of each model (Fig. 4). The DRD and smooth-DRD models were both similar to ARD in that they achieved sparsity by shrinking the prior variance of unnecessary coefficients to zero. However, unlike ARD, their inferred prior covariances both exhibited clusters of non-zero coefficients, reflecting the dependencies introduced by the latent Gaussian process. Note also that ARD and DRD covariances were both diagonal, making the weights independent given the prior variances, whereas the smooth-DRD covariance had off-diagonal structure that induced smoothness.

To quantitatively validate that our estimators succeed at identifying structured sparse and smooth structure, we performed simulated experiments using data drawn from the DRD generative model. For each experiment, we generated simulated data with $n = 500$ samples from a p-element weight vector, and varied p from 500 to 4000. We used hyperparameters GP mean $b = -8$, GP length scale $l = p/40$, GP marginal variance $\rho = 36$, smoothness length scale $\delta = p/20$, and varied measurement noise variance σ^2 between 1 and 50. The sparsity ratio for the sampled weights $\mathbf{w}$ was approximately 0.20, where we considered

[2] `http://nilearn.github.io/index.html`

[3] `https://github.com/Hua-Zhou/SparseReg`

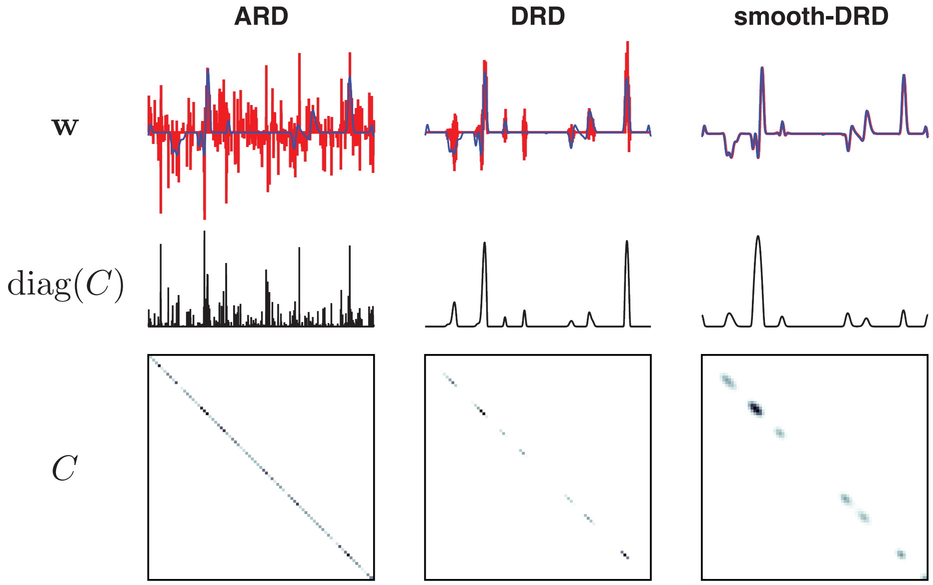

Figure 4: Estimated filter weights and prior covariances. The upper row shows the true filter (blue) and estimated ones (red); the middle row displays the diagonal of each estimated covariance matrix; and the bottom row shows the entire estimated covariance matrix for each prior.

weights with $|w_i| > 0.005$ to be non-zero. We varied training set size from $n = 100$ to 400 and kept a fixed test size of 100 samples. (We noted that even with $n = 400$ samples, the problem resides in the $n < p$ small-sample regime). We repeated each experiment 5 times.

We compared performance of DRD estimators to the above-mentioned estimators. Fig. 5 shows the reconstruction performance (R^2) of the true regression weights $\mathbf{w}$ for different estimators as a function of noise variance, training set size and dimension.

We found that Laplace and MCMC estimates for the smooth-DRD model outperformed other estimators and were approximately equally accurate, indicating that use of Laplace approximation did not noticeably harm performance relative to the fully Bayesian estimate. ASD had a good performance indicating that for these extremely smooth weights, smoothness was a more useful form of regularization than structured sparsity conferred by DRD. DRD models came next. DRD-MCMC was slightly better due to the robustness of the fully Bayesian inference. DRD-Laplace-exp and DRD-Convex employed the exponential nonlinearity when transforming $\mathbf{u}$ to the diagonal of the covariance matrix. DRD-Laplace-rec used a soft-rectifier nonlinearity which was more numerically stable. They had similar R^2 values with TV-L1 when recovering $\mathbf{w}$, but were better than all the other methods. We can also investigate the influence of each variable, i.e. noise variance, training size or dimension. When increasing the noise variance, all the R^2 values dropped; smooth-DRD-MCMC outperformed others with $\sigma^2 = 50$ indicating the power of the fully Bayesian estimate and the nontrivial effect of simultaneously imposing local sparsity and smoothness. When increasing the training size, DRD-Laplace models and DRD-Convex outperformed TV-L1

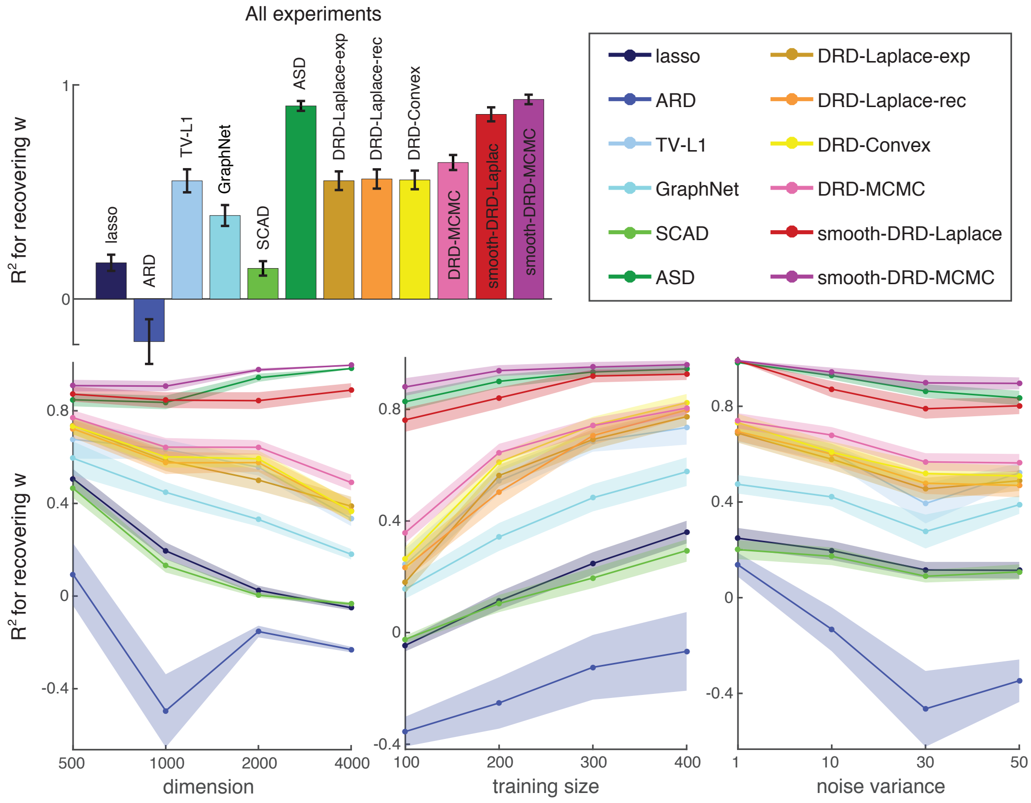

Figure 5: Comparison of performance recovering true regression weights $\mathbf{w}$ in simulated experiments as a function of dimensions p (lower left), number of samples n (lower middle), and noise variance σ^2 (lower right). Experiments were repeated five times for each of 64 combinatorial settings of four values for p, n, and σ^2. Traces show average R^2 (±1 standard error of the mean (SEM)) as a function of each variable, and the bar plot (top row) shows average R^2 (±1 SEM) over all $5 \times 64 = 320$ experiments.

and were comparable with DRD-MCMC, which was due to the decreasing optimizing complexity. Also surprisingly, smooth-DRD estimators achieved nearly perfect reconstruction performance over all the training sizes and all the dimensions.

Fig. 6 shows the R^2 performance for regression prediction on the test set for different estimators as a function of noise variance, training set size and dimension. The reconstruction performance for recovering the true $\mathbf{y}_{test}$ is given by $R^2 = 1 - \frac{||\mathbf{y}_{test} - \hat{\mathbf{y}}_{test}||_2^2}{||\mathbf{y}_{test} - \bar{\mathbf{y}}_{test}||_2^2}$, where $\bar{\mathbf{y}}_{test} = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} \mathbf{y}_{test,i}$ is the mean of vector $\mathbf{y}_{test}$. The top-left subfigure presents the averaged R^2 values and the confidence intervals for $\hat{\mathbf{y}}_{test}$ over all runs. Similar to R^2 for $\mathbf{w}$, ASD estimate, Laplace and MCMC estimates for the smooth-DRD model outperformed other estimators.

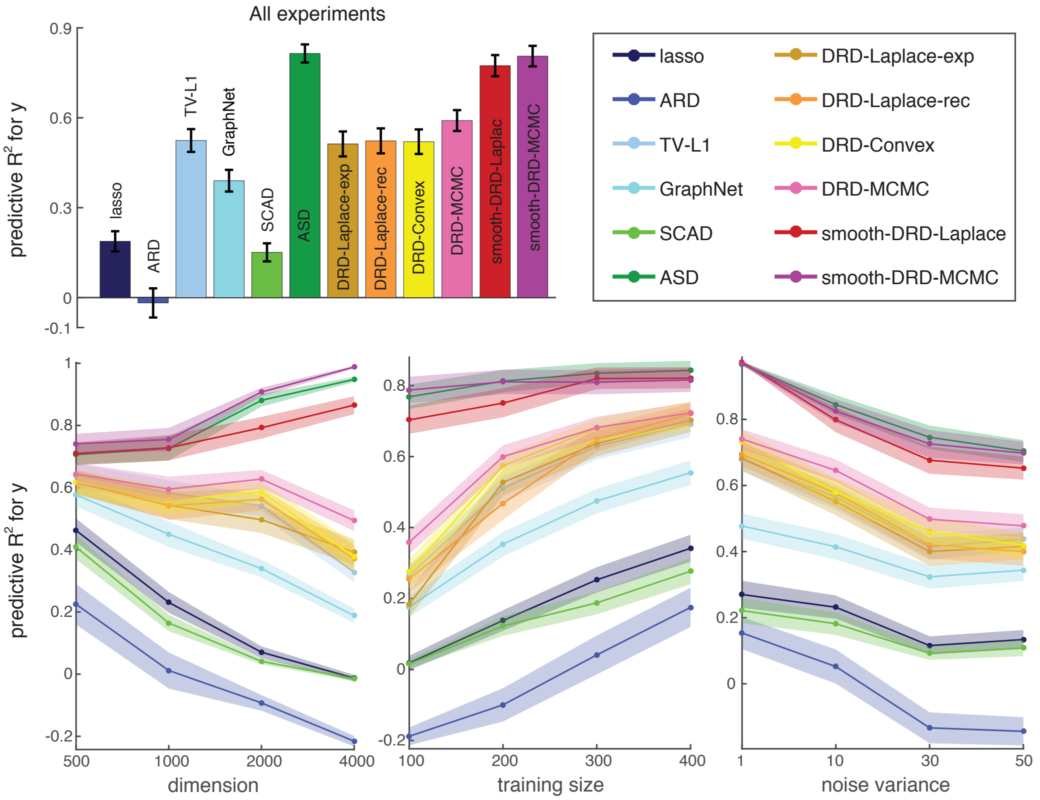

Figure 6: Comparison of performance predicting held-out responses $\mathbf{y}_{test}$ in simulated experiments as a function of dimensions p (lower left), number of samples n (lower middle), and noise variance σ^2 (lower right). Traces show average R^2 (±1 SEM), and the bar plot (top row) shows average R^2 (±1 SEM) over all experiments. Simulation experiments were the same as in Fig. 5.

Fig. 7 shows the AUC (Area Under the receiver operator characteristic Curve) values for different estimators as a function of noise variance, training set size and dimension. The AUC metric quantifies accuracy in recovering the binary support for $\mathbf{w}$, which is useful for assessing the effects of structured sparsity. For this metric, the smooth DRD estimators outperformed other methods, and the ASD estimator performed much worse due to its lack of sparsity. The Laplace approximation based DRD models performed slightly better than DRD-MCMC because the sparsity of MCMC estimates was diluted by averaging across multiple samples.

Overall, smooth-DRD outperformed all other methods using all metrics. This shows that combining sparsity and smoothness can provide major advantages over methods that impose only one or the other. This flexible framework for integrating structured sparsity and smoothness is one of the primary contributions of our work, in contrast to previous methods

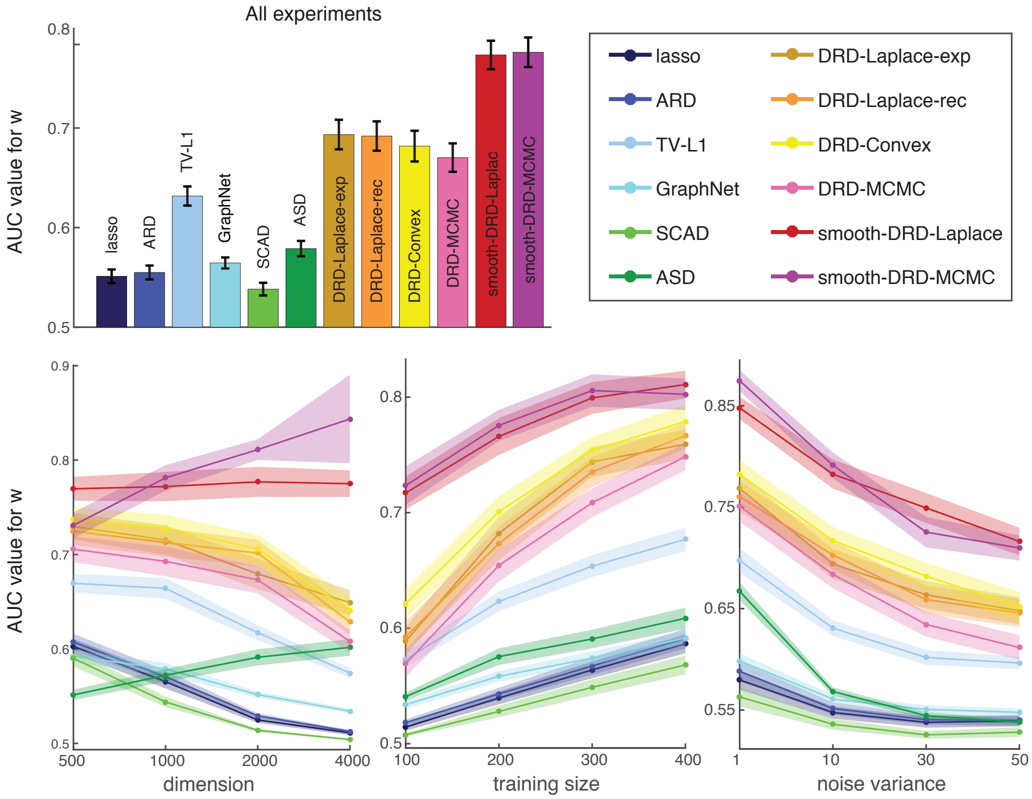

Figure 7: Comparison of performance at recovering support of regression weights in simulated experiments, quantified with area under the ROC curve (AUC), as a function of dimensions p (lower left), number of samples n (lower middle), and noise variance σ^2 (lower right). Traces show average AUC (± 1 SEM), and the bar plot (top row) shows average AUC (± 1 SEM) over all experiments. Simulation experiments were the same as in Fig. 5. Support recovery was quantified by taking all coefficients $|\mathbf{w}_i| > 0.005$ as non-zero.

in the structured sparsity literature which consider only sparsity. The code and simulated results are available online[4].

5.2. Computational complexity and optimization

We have described two basic approaches to inference for DRD: evidence optimization using the Laplace approximation and MCMC sampling-based inference. The main computational difficulty associated with Laplace-based methods is the Hessian matrix, which provides the precision matrix for the approximate Gaussian posterior distribution. This matrix costs $O(p^2)$ to store and contributes $O(p^3)$ time complexity for computation of the log-determinant. We can reduce these costs to $O(p_f^2)$ storage and $O(p_f^3)$ time, where $p_f < p$

[4] `https://github.com/waq1129/DRD.git`

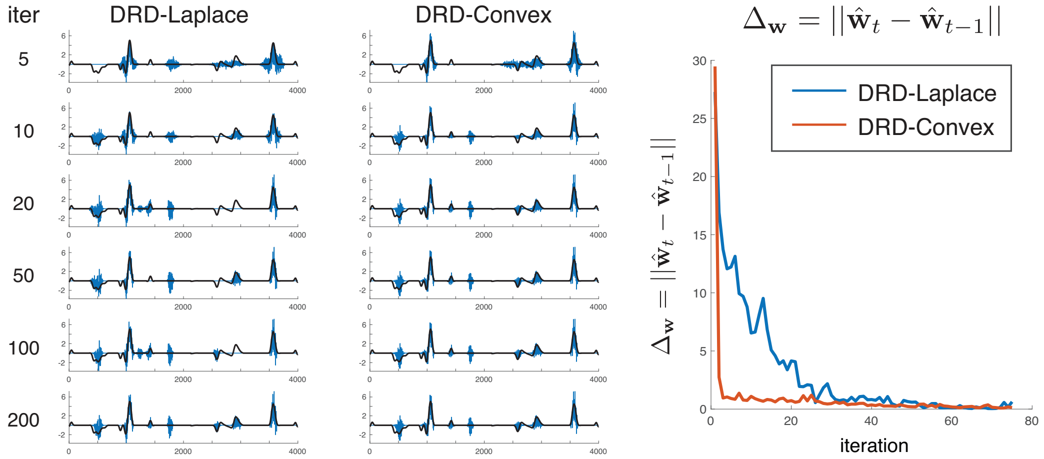

Figure 8: Comparison of the optimization of weights $\hat{\mathbf{w}}$ between DRD-Laplace and DRD-Convex. The first two columns show the weights obtained after 5, 10, 20, 50, 100, and 200 iterations with the same initialization under the two estimators, with true weights indicated in black. The third column shows the change in weights after each iteration of the standard and convex optimization algorithms over the first 80 iterations, showing that the convex algorithm made much smaller adjustments to the weights after the first few iterations and thus converged more rapidly.

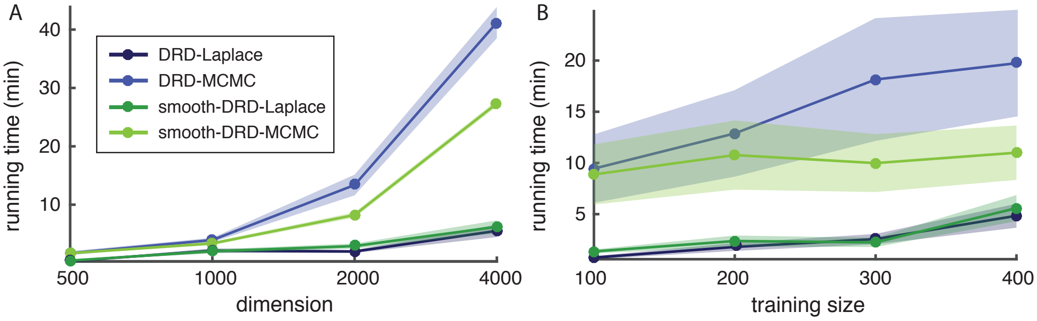

Figure 9: Running times for DRD estimators as a function of dimensions p (left) and number of samples n (right). Each point is an average across 20 simulated experiments, and the shaded area represents ±1 SEM. For MCMC, running time was determined by the time to collect 100 posterior samples after burn-in.

is the number of non-zero Fourier coefficients in the spectral domain representation of the latent Gaussian process. This savings can be significant in problems with strong region sparsity, that is, when the zero coefficients arise in large contiguous blocks. However, in very high dimensions, the Laplace based methods may be practically infeasible due to the impossibility of storing the Hessian.

We also described a two-step convex relaxation of the Laplace method (DRD-Convex), which takes more time per iteration than the standard Laplace method (DRD-Laplace) due to the need for a two-step optimization procedure. However, we find that the DRD-Convex takes fewer iterations to converge (see Fig. 8), and in some cases proves more successful at avoiding sub-optimal local optima.

The MCMC-based inference has a time complexity of only $O(n^2 p_f)$ per sample, due to the fact that there is no need to compute the Hessian. However, MCMC-based inference is typically slower due to the need for a burn-in period and the generation of many samples from the posterior. Fig. 9 shows a comparison of running time for the two inference methods for both DRD and smooth-DRD models. For the Laplace method, we used a stopping criterion that the change in $\mathbf{w}$ was less than 0.0001. For the MCMC method, we assessed burn-in using a criterion on the relative change in $\mathbf{w}$, and then collected 100 posterior samples. Inference for the smooth-DRD model was faster than for standard DRD due to the fact that the smoothing prior effectively prunes high frequencies, reducing the dimensionality of the search space for $\mathbf{w}$. In these experiments, increasing dimension p elicited larger increases in computation time than increasing training set size n.

6. Phase transition in sparse signal recovery

Compressive sensing focuses on the recovery of sparse high-dimensional signals in settings where the number of signal coefficients p exceeds the number of measurements n. Recent work has shown that the recovery of sparse signals exhibits a phase transition between perfect and imperfect recovery as a function of the number of measurements (Ganguli and Sompolinsky, 2010; Amelunxen et al., 2014). Namely, when the measurement fraction $\gamma = n/p$ exceeds some critical value that depends on signal sparsity, the signal can be recovered perfectly with probability approaching 1, whereas for γ below this value, estimates contain errors with probability approaching 1. However, these results were derived for the case where non-zero coefficients are randomly located within the signal vector. Here we show that DRD can obtain dramatic improvements over the phase transition curve for *iid* sparse signals when the non-zero coefficients arise in clusters.

We performed simulated experiments to examine the effects of group structure on the empirical phase transition between perfect and imperfect recovery of sparse signals. The measurement equation is given by the noiseless version of the linear system we have considered so far: $\mathbf{y} = X\mathbf{w}$, where $\mathbf{w} \in \mathbb{R}^p$ is the sparse signal, $\mathbf{y} \in \mathbb{R}^n$ is the (dense) measurement vector, and here $X \in \mathbb{R}^{n \times p}$ is a (short, fat) random measurement matrix with entries drawn *iid* from a standard normal distribution. We define the sparsity of the signal as $\alpha = k/p$, where k is the number of non-zero signal coefficients in $\mathbf{w}$.

To explore the effects of group structure, we considered the signal coefficients in $\mathbf{w}$ to have 1D spatial structure and introduced a parameter g specifying the number of spatial groups or clusters into which the non-zero coefficients were divided. When $g = 1$, the non-zero coefficients formed a single contiguous block of length k, with location uniformly distributed within $\mathbf{w}$. When $g > 2$, the non-zero coefficients were divided into g blocks of size k/g, and the locations of these blocks were uniformly distributed within $\mathbf{w}$ subject to

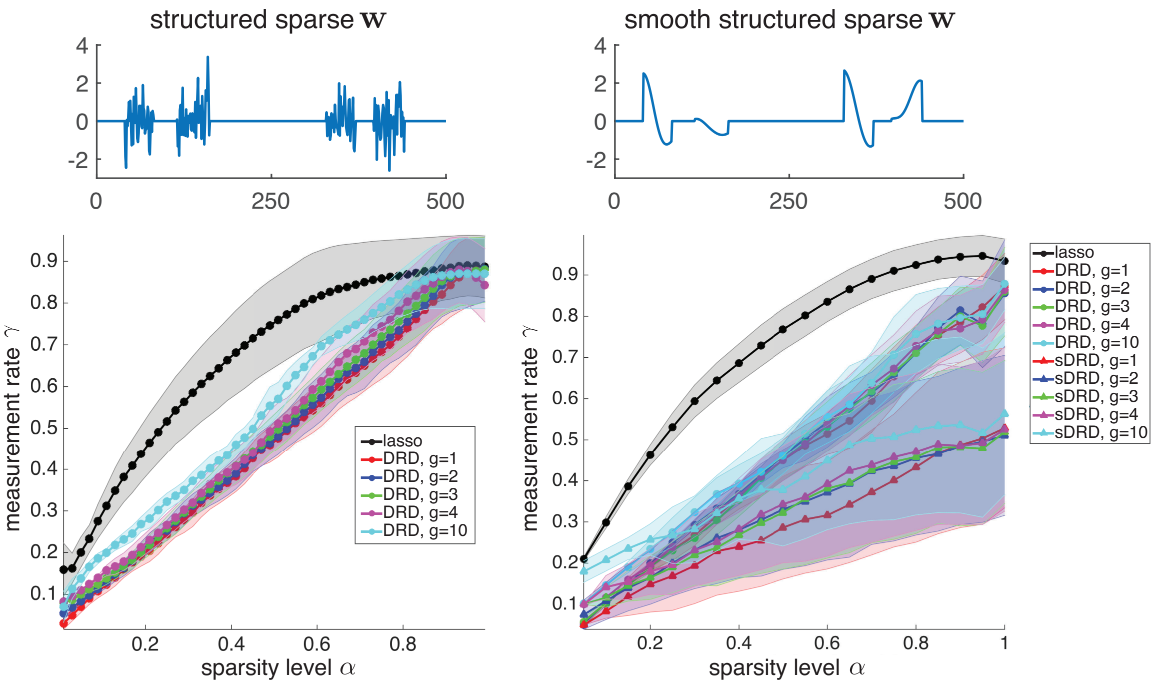

Figure 10: **Phase transitions for DRD and smooth-DRD (sDRD) estimators on signals with structured sparsity.** Top row shows example signals $\mathbf{w}$ of dimensions $p = 500$, which contain randomly positioned blocks of non-zero coefficients. Non-zero coefficients were clustered into varying numbers of groups g, and drawn either iid from a standard normal distribution (left column) or from a Gaussian with a smoothing kernel (length scale was 20) (right column), to illustrate the effects of smoothness. To compute phase transition curves, we analyzed the recovery behavior of each estimator at every point in a 2D grid of sparsity levels α and measurement rates γ. At each point, we generated 10 random signals $\mathbf{w}$, projected them noiselessly onto a random Gaussian measurement matrix X, and computed lasso and DRD estimates $\hat{\mathbf{w}}$. We then calculated the R^2 value of the estimates for each trial at every grid point (α, γ). An estimator was considered to achieve perfect recovery if all 10 trials resulted in $R^2 > 0.95$, and perfect failure if all 10 trials resulted in $R^2 \leq 0.95$; remaining points were considered to fall in the phase transition region. For each estimator, the shaded region indicates the phase transition region, and solid line indicates its center of mass along the y-axis.

the constraint that blocks remained disjoint. Once the sparsity pattern was determined, we sampled the non-zero coefficients from a standard normal distribution.

Fig. 10 shows the empirical phase transition curves for lasso and DRD estimators for sparse signals with non-zero coefficients clustered into varying numbers of groups g. These curves show the boundary between perfect and imperfect signal recovery for different estimators in the 2D space of signal sparsity level α and measurement fraction γ. The left bottom plot shows that DRD estimators achieved perfect signal recovery for much lower measurement rates, even when non-zero coefficients were clustered into as many as 10 groups. Here DRD achieved transition to perfect recovery along the main diagonal, whereas lasso exhibited an arc-shape transition curve described previously (Ganguli and Sompolinsky, 2010; Amelunxen et al., 2014), indicating that more measurements were required to recovery signals of equal sparsity. In the right bottom plot, we generated the non-zero coefficients from a Gaussian distribution with a smoothing kernel whose length scale equaled to 20, so that non-zero coefficients were smooth as well as sparse. In these plots, we compared lasso estimates (which do not benefit from group or smooth structure) to standard DRD and smooth-DRD estimates. This reveals that smoothness allows for further reductions in measurement rates, with perfect signal recovery achievable well below that of the non-smooth DRD estimates.

7. Applications to brain imaging data

Functional magnetic resonance imaging (fMRI) measures blood oxygenation levels, which provide a proxy for neural activity in different parts of the brain. Although these measurements are noisy and indirect, fMRI is one of the primary non-invasive methods for measuring activity in human brains, and it has provided insight into the neural basis for a wide variety of cognitive abilities and functional pathologies.

A primary problem of interest in the fMRI literature is “decoding”, which involves the use of linear classification and regression methods to identify the stimulus or behavior associated with measured brain activity. Decoding is a challenging statistical problem because the number of volumetric pixels or “voxels” measured with fMRI is typically far greater than the number of trials in an experiment; a full brain volume typically contains 50K voxels whereas most experiments produce only a few hundred observations.

Standard approaches to decoding have therefore tended to exploit sparsity, corresponding to the assumption that only a small set of brain voxels are relevant for decoding a particular set of stimuli (Carroll et al., 2009). However, the set of voxels useful for a specific decoding task are not randomly distributed throughout the brain, but tend to arise in clusters; if one voxel carries information useful for decoding, it is *a priori* likely that nearby voxels do too, given that voxels represent an arbitrary discretization of continuous underlying brain structures. We therefore explored brain decoding as an ideal application for evaluating the performance of our estimators.

7.1. Gambling task

We first considered the regression problem of decoding gains and losses from fMRI measurements recorded in a gambling task (Tom et al., 2007, 2011). In this experiment, event-related fMRI was administered while healthy human participants performed a decision-making task. In each trial, a gamble with a potential gain and loss (each with 50% probability) was pre-

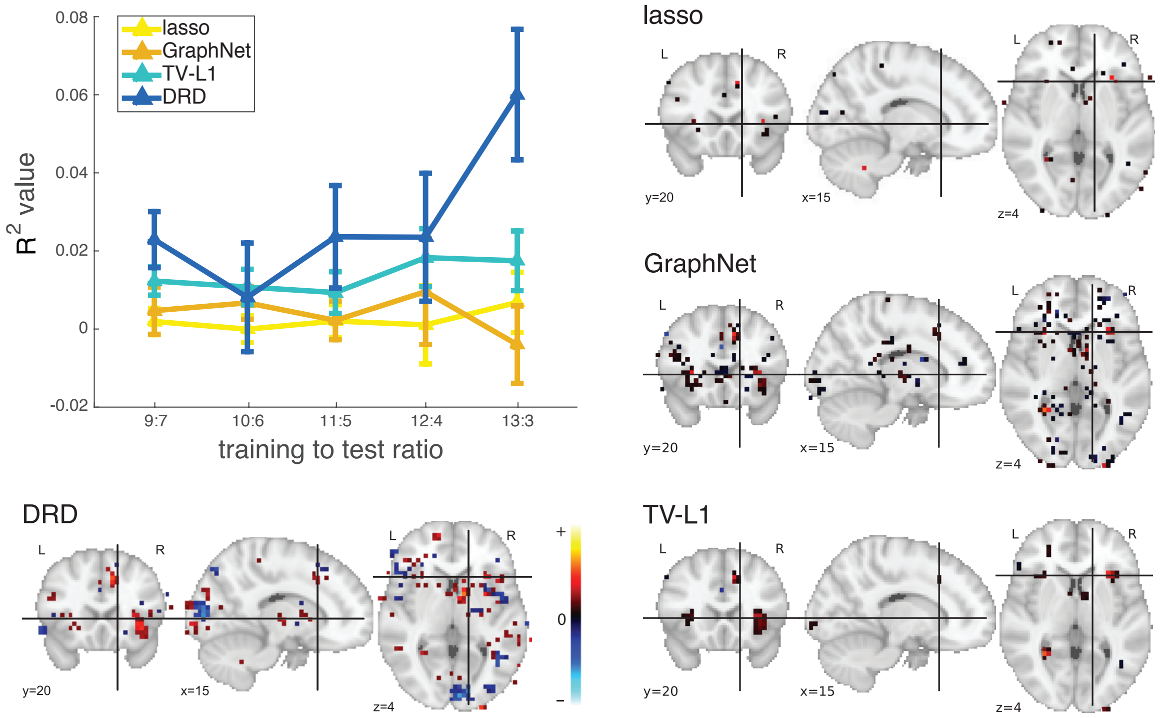

Figure 11: Top left panel shows average test R^2 values on the gambling dataset as a function of the train-test ratio for lasso, GraphNet, TV-L1 and DRD. The x-axis is the train-test split ratio and the y-axis is the R^2 criterion. The remaining panels show the estimated fMRI weight maps, overlaid on a structural fMRI image. Colors indicate the sign and magnitude of the weights (see color bar, red for positive, blue for negative, black for small). The DRD figure was obtained by cutting off small weight coefficients with a small threshold value at 0.004 (about 12% of the maximal absolute coefficient value).

sented for 3s, and the participants were instructed to decide whether to accept or reject the gamble. Experimenters varied amount of the potential gain and loss across trials. The regression task is to predict the gain of the gamble from the fMRI images recorded during the decision-making task.

After standard preprocessing, the regression dataset consisted of 16 subjects with 48 fMRI measurements per subject (resulting from 6 repeated presentations of 8 different gambles). fMRI measurements were obtained from a 3D brain volume of $40 \times 48 \times 34$ voxels, each of size $4 \times 4 \times 4$ mm, from which a subset of 33,177 valid brain voxels were used for analysis. The full dataset of 16 subjects therefore contained $n = 768$ samples in a $p = 33,177$ dimensional space.

We evaluated inter-subject prediction performance by estimating regression weights with data from a subset of the 16 subjects, and computing prediction accuracy for data from held-out subjects. To assess the performance, we varied the train-test ratio in number of subjects from 9:7 to 13:3. We performed 10 different random train-test splits for each ratio. We used 5-fold cross-validation to set hyperparameters for all models, including DRD. For DRD models, the Laplace approximation was intractable due to the high dimensionality of the weight vector ($p = 33,177$). We therefore computed MAP estimates of the latent vector $\mathbf{u}$ conditioned on the hyperparameters, and set hyperparameters using cross-validation.

The curves in the top left panel of Fig. 11 show the performance of lasso, GraphNet, TV-L1 and DRD estimators. We found that DRD outperformed other estimators at nearly all train-test ratios, with a noticeable advantage at the largest training set size. However, we noted that the SNR of this dataset was low, making inter-subject prediction difficult and resulting in low accuracy for all methods. A non-trivial preprocessing stage, such as hyper-alignment (Chen et al., 2015), could be used to map different subjects into a shared subspace, which reduces the low SNR induced by inter-subject variability and could possibly improve performance.

Fig. 11 also shows the inferred regression weights for each estimator. The GraphNet and lasso weights had high sparsity, presumably due to the low SNR of the dataset, while TV-L1 weights exhibited small blocks of non-zero coefficients with constant value within each block, consistent with the structure expected for the TV-L1 penalty. The DRD weights were not sparse in a strict L0 sense, due to the fact that sparsity arises from soft-rectification of negative latents governing the prior variance. We therefore thresholded DRD weights for plotting purposes, revealing that the weights contributing most to prediction performance tended to cluster, as expected, although weights within each cluster were not constant. One noteworthy observation is that DRD estimate had positive (red) as well as negative weights (blue), while other estimates were largely devoid of regions with negative weights. Note that voxels in black indicate weights close to zero, which therefore contributed relatively little to readout.

7.2. Age prediction task

Next we considered the problem of predicting a subject's age from a measured map of gray-matter concentration, using data from the Open Access Series of Imaging Studies (OASIS) (Marcus et al., 2007). The OASIS dataset consisted of T1-weighted MRI scans data from 403 subjects aged 18 to 96, with 3 or 4 scans per subject. One hundred of these subjects were over 60 years of age and had been clinically diagnosed with Alzheimer's. The repeated scans provided high signal-to-noise ratio, making the dataset feasible for inter-subject analyses.

A natural regression problem for this dataset is to predict the age of subject from their anatomical MRI data. The full dataset consisted of 403 samples with a $91 \times 109 \times 91$ 3D volume and 129,081 valid voxels. To assess the performance, we varied the training ratio from 0.4 to 0.8 out of the 403 subjects, and averaged over 5 random splits for each ratio.

The curves in the top left panel of Fig. 12 show mean absolute errors between the true age and the predicted age for each estimator, evaluated on test data. The DRD and smooth-DRD estimators, which performed similarly well, achieved lower error than lasso, GraphNet, and TV-L1 estimators. The inferred regression weights (Fig. 12) reveal that the most

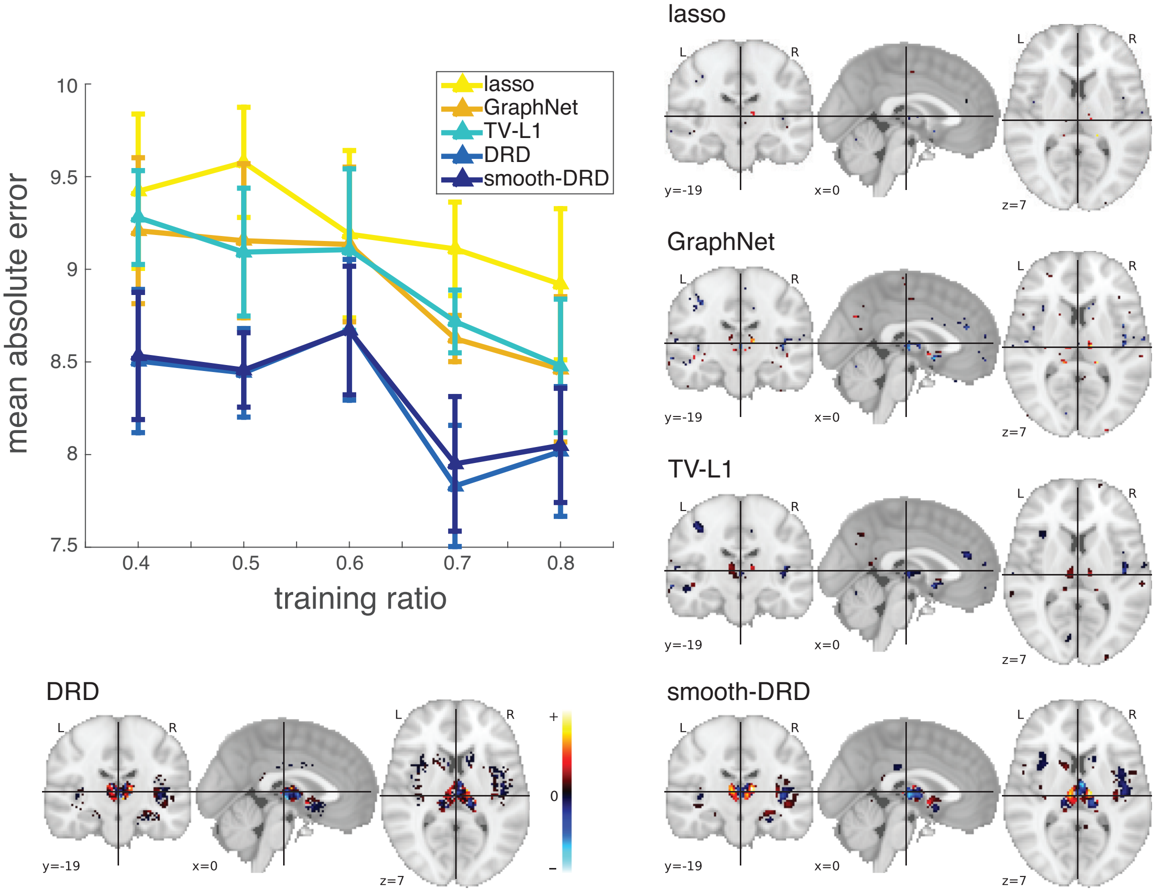

Figure 12: Top left panel shows average test mean absolute error values on the age regression dataset as a function of the training ratio for lasso, GraphNet, TV-L1, DRD and smooth-DRD. The x-axis is the ratio of the training data over the entire dataset and the y-axis is the mean absolute error criterion. The remaining panels show the estimated MRI weight maps, overlaid on a structural MRI image. Colors indicate the sign and magnitude of the weights (see color bar).

informative voxels for predicting age lie in thalamus and the basal ganglia; this is consistent with previous findings about the relationship between Alzheimer's disease progression and anatomical changes in gray matter (de Jong et al., 2008; Cho et al., 2014). Thalamus and basal ganglia were more clearly highlighted in the inferred DRD and smooth-DRD regression weights, which contained larger and more concentrated regions around these two structures. The smooth-DRD weights exhibited stronger spatial clustering than DRD weights, although regression performance was not noticeably different between the two estimators.

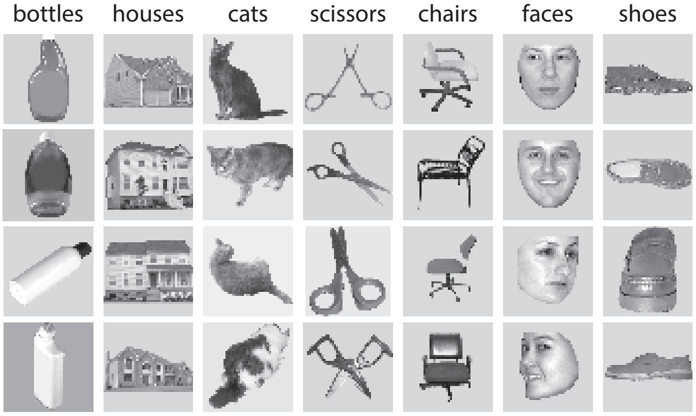

Figure 13: Examples of the stimuli for 7 categories (except for scrambled control images).

7.3. Visual recognition task

In a third application, we examined the problem of decoding faces and objects from fMRI measurements during a visual recognition task. We used a popular fMRI dataset from a study on face and object representation in human ventral temporal cortex (Haxby et al., 2001). In this experiment, 6 subjects were asked to recognize 8 different types of objects (bottles, houses, cats, scissors, chairs, faces, shoes and scrambled control images, examples in Fig. 13). Each subject participated 12 sessions of experiment. In each session, the subjects viewed images of eight object categories, with 9 full-brain measurements per category.

We assessed performance by training linear classifiers to discriminate between pairs of objects, e.g., face vs. bottle, for each of the 28 possible binary classifications among 8 objects. We trained the weights $\mathbf{w}$ for each model to linearly map fMRI measurements $\mathbf{x}$ to binary labels $y \in \{-1, +1\}$ by minimizing squared error. Note that a Bernoulli log-likelihood or logistic loss would be more appropriate for this binary classification task, but we used squared error loss because it allows for analytic marginalization over weights. We assessed decoding accuracy on test data using predicted labels $\hat{y} = \text{sign}(\mathbf{w}^\top \mathbf{x})$. We divided 12 sessions of data per subject into train-test splits of 5:7, 6:6 and 7:5. When training with data from N sessions, the training dataset consisted of 18N full-brain measurements (9 measurements per category × 2 categories). Each measurement contained 24,083 valid voxels from a $40 \times 64 \times 64$ 3D volume.

Fig. 14 shows the classification performance of lasso, GraphNet, TV-L1, DRD, and smooth-DRD estimators, averaged over 6 subjects and across the three train-test splits. The smooth-DRD estimate achieved the highest accuracy for most of the binary classifications, while the DRD estimate achieved second highest accuracy. The left column in Fig. 15 shows the regression weights estimated for the house vs. bottle classification task. The DRD and smooth-DRD weights both had significant positive regions in the parahippocampal place area (PPA), an area known to respond to images of places (Epstein et al., 1999) and negative regions in the lateral occipital complex (LOC), an area known to respond to objects (Eger

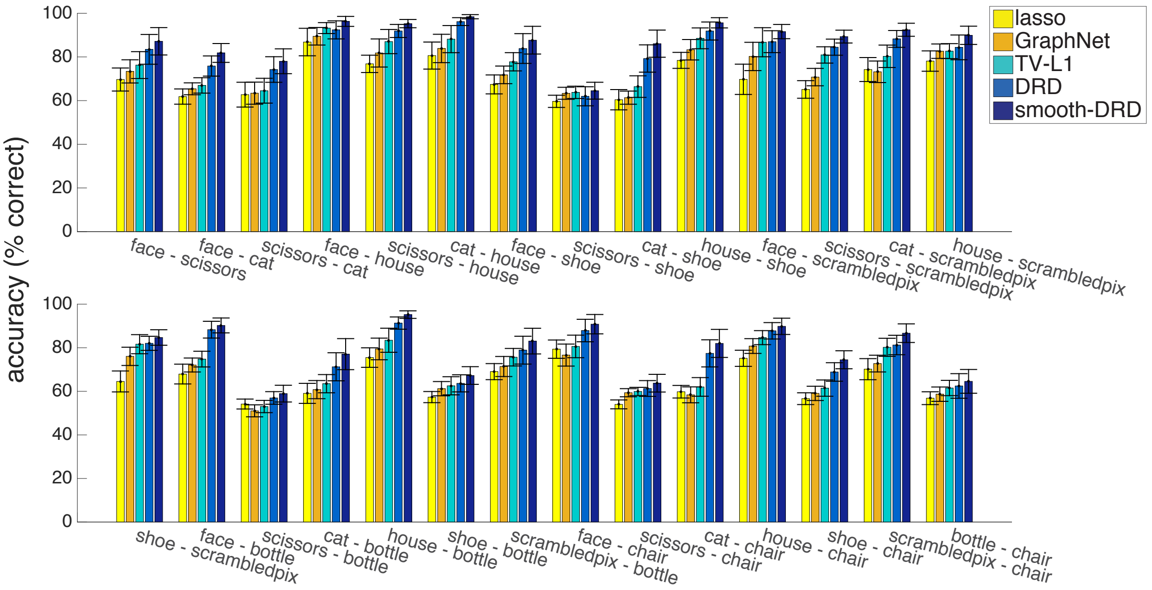

Figure 14: Classification accuracy performance for lasso, GraphNet, TV-L1, DRD and smooth-DRD, averaged across 6 subjects and three different train-test splits (5:7, 6:6 and 7:5). Error bars indicate ±1 SEM, averaged over train-test splits. The x-axis labels indicate pairs of object categories considered for binary classification.

et al., 2008). By comparison, TV-L1 and GraphNet weights in LOC were neither strong nor clustered. The right column in Fig. 15 shows regression weights for the face vs. scrambled-pixels classification task. All methods managed to discover active responses around LOC and fusiform face area (FFA) (specialized for facial recognition) (Kanwisher et al., 1997), though DRD and smooth-DRD weights exhibited fewer isolated non-zero weights in areas far from the temporal and occipital lobes.

8. Discussion

In this paper, we introduced dependent relevance determination (DRD), a hierarchical Bayesian model for sparse, localized, and smooth regression weights. This model is appropriate for regression settings in which the regressors can be arranged geometrically as a vector, matrix, or tensor, and exhibit local dependencies within this structure (e.g., tensors of 3D brain measurements).

The DRD model takes its inspiration from the automatic relevance determination (ARD) model (Tipping, 2001), but adds a Gaussian process to introduce dependencies between prior variances of regression weights as a function of distance between their regressors. Samples from the DRD prior therefore exhibit clustering of non-zero weights. However, weights

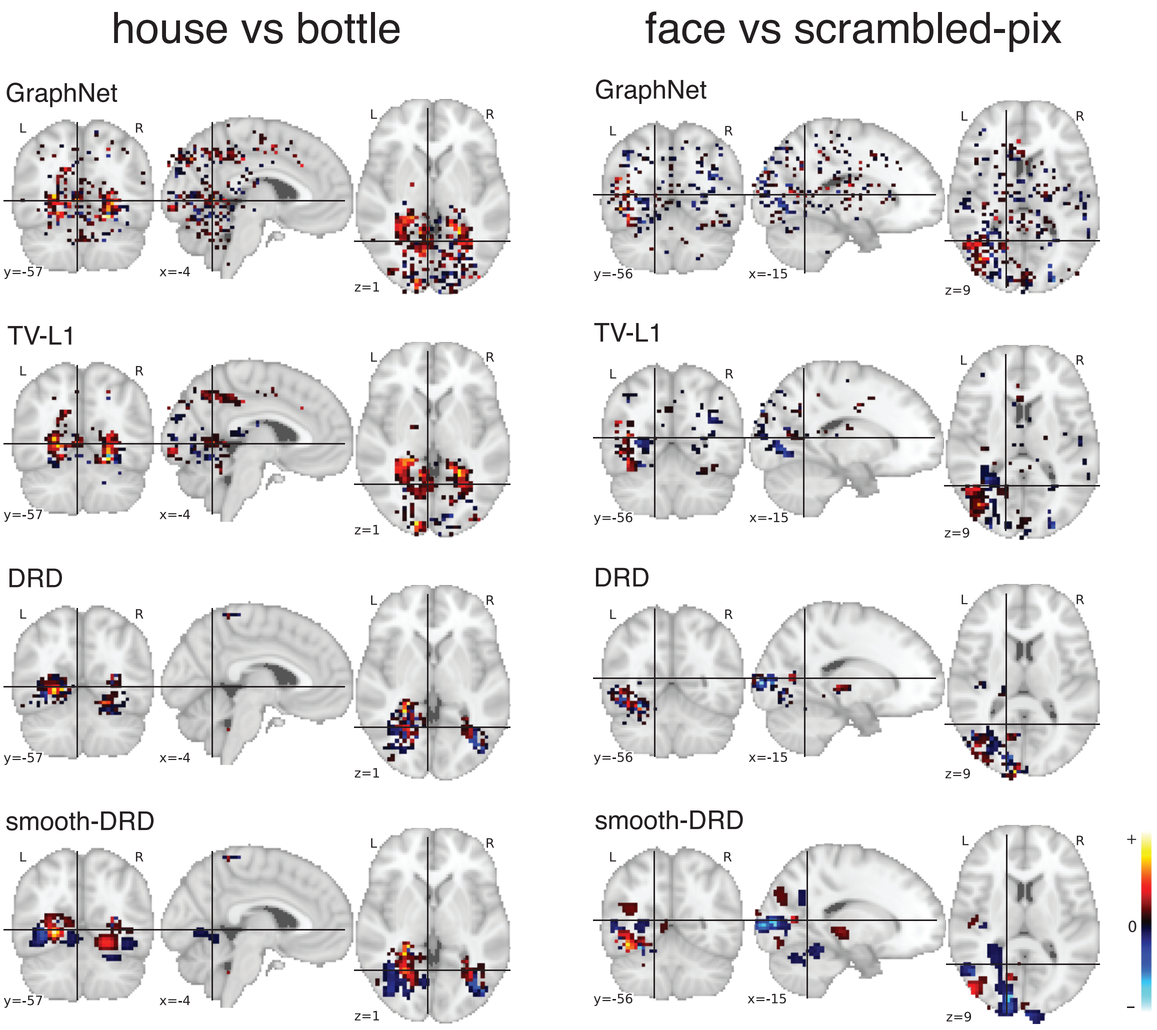

Figure 15: **Left column**: weight maps for the house vs bottle pair. **Right column**: weight maps for the face vs scrambled-pix pair. The methods shown are GraphNet, TV-L1, DRD and smooth-DRD.

sampled from the standard DRD prior are uncorrelated, meaning there is no tendency for such weight tensors to be smooth. For this reason, we introduced the smooth-DRD model, which composes the standard DRD prior with a smoothness-inducing covariance function (Sahani and Linden, 2003; Park and Pillow, 2011). Weights sampled from the resulting model tend to be sparse as well as smooth, with islands of smooth, non-zero weight features surrounded by oceans of zeros.

We described two methods for inferring the model parameters: one based on the Laplace approximation and a second based on MCMC. We proposed a novel variant of the Laplace based approach involving a two-stage convex relaxation of the log posterior.

Lastly, we carried out simulated and real application experiments to compare DRD with a variety of other methods, including lasso, SCAD, GraphNet, TV-L1, and etc. For an L_2 loss, a convex L_1 penalty leads to strong amplitude bias for lasso, while a non-convex penalty, e.g., SCAD, can overcome the strong bias. Lasso could underfit in order to get a tight support due to incorrect amplitudes of the coefficients when doing cross-validation using a L_2-based criterion. Therefore, we included SCAD apart from lasso and employed AUC as the metric for support identification. The synthetic experiments generated true weight vectors from the proposed generative model, thus aiming at validating the proposed model. We also examined phase transitions between perfect and imperfect recovery using data from a generating model different from DRD, showing that the DRD and smooth-DRD model could achieve perfect recovery with far fewer measurements when the non-zero weights in a signal were clustered. We further applied our models to three real-world brain imaging datasets. We found that DRD and smooth DRD achieved better prediction performance than previous methods, while also achieving high interpretability with regression/classification weights defined by smooth, clustered sets of voxels. Note that for the final fMRI decoding application task, a Bernoulli noise model (corresponding to a logistic loss function) would have been more appropriate than the Gaussian noise model we assumed, due to the binary nature of the outputs. Gaussian noise is a useful modeling assumption for the DRD model because it yields an analytical expression for the conditional evidence (eq. 14), which can be directly optimized for the latent process governing region sparsity. An important direction for future research will therefore be to extend DRD to incorporate Bernoulli and other likelihoods. Such extensions will need to use approximate inference methods or sampling to compute the integral over regression weights (eq. 13), but there is no conceptual barrier to incorporating region sparsity into models with binary and other non-Gaussian outputs.

The DRD and smooth-DRD models offer a powerful statistical framework for attacking problems in which sparsity is overlaid with local dependencies, a scenario that arises commonly in (for example) spatial and temporal regression problems. Recent work has shown successful application of a closely related model for capturing dependencies between sparse variables in genomic data (Engelhardt and Adams, 2014). In future work, we expect the DRD framework to find applications beyond the regression/classification setting, including structured latent factor models (Chen et al., 2015) and false discovery rate estimation (Tansey et al., 2017).

Acknowledgments

This work was supported by grants from the Sloan Foundation (JWP), McKnight Foundation (JWP), Simons Collaboration on the Global Brain (SCGB AWD1004351, JWP), and the NSF CAREER Award (IIS-1150186, JWP).

References

Alexandre Abraham, Fabian Pedregosa, Michael Eickenberg, Philippe Gervais, Andreas Mueller, Jean Kossaifi, Alexandre Gramfort, Bertrand Thirion, and Gaël Varoquaux.

Machine learning for neuroimaging with scikit-learn. *Frontiers in neuroinformatics*, 8:14, 2014.

Dennis Amelunxen, Martin Lotz, Michael B McCoy, and Joel A Tropp. Living on the edge: Phase transitions in convex programs with random data. *Information and Inference*, page iau005, 2014.

Michael R Andersen, Ole Winther, and Lars K Hansen. Bayesian inference for structured spike and slab priors. In *Advances in Neural Information Processing Systems*, pages 1745–1753, 2014.

Michael Riis Andersen, Aki Vehtari, Ole Winther, and Lars Kai Hansen. Bayesian inference for spatio-temporal spike and slab priors. *arXiv preprint arXiv:1509.04752*, 2015.

Luca Baldassarre, Janaina Mourao-Miranda, and Massimiliano Pontil. Structured sparsity models for brain decoding from fmri data. In *Pattern Recognition in NeuroImaging (PRNI), 2012 International Workshop on*, pages 5–8. IEEE, 2012.

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.

Carlos M Carvalho, Nicholas G Polson, and James G Scott. Handling sparsity via the horseshoe. In *International Conference on Artificial Intelligence and Statistics*, pages 73–80, 2009.

Po-Hsuan Cameron Chen, Janice Chen, Yaara Yeshurun, Uri Hasson, James Haxby, and Peter J Ramadge. A reduced-dimension fmri shared response model. In *Advances in Neural Information Processing Systems*, pages 460–468, 2015.

Hanna Cho, Jeong-Hun Kim, Changsoo Kim, Byoung Seok Ye, Hee Jin Kim, Cindy W Yoon, Young Noh, Geon Ha Kim, Yeo Jin Kim, Jung-Hyun Kim, et al. Shape changes of the basal ganglia and thalamus in alzheimer's disease: a three-year longitudinal study. *Journal of Alzheimer's Disease*, 40(2):285–295, 2014.

Laura W de Jong, Karin van der Hiele, Ilya M Veer, Jeanine Houwing, Rudi Westendorp, Elem Bollen, Paul W de Bruin, Huub Middelkoop, Mark A van Buchem, and Jeroen van der Grond. Strongly reduced volumes of putamen and thalamus in alzheimer's disease: an mri study. *Brain*, 131(12):3277–3285, 2008.

Evelyn Eger, John Ashburner, John-Dylan Haynes, Raymond J Dolan, and Geraint Rees. fmri activity patterns in human loc carry information about object exemplars within category. *Journal of cognitive neuroscience*, 20(2):356–370, 2008.

Barbara E Engelhardt and Ryan P Adams. Bayesian structured sparsity from gaussian fields. *arXiv preprint arXiv:1407.2235*, 2014.

Russell Epstein, Alison Harris, Damian Stanley, and Nancy Kanwisher. The parahippocampal place area: Recognition, navigation, or encoding? *Neuron*, 23(1):115–125, 1999.

Jianqing Fan and Runze Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360, 2001.

Anita C Faul and Michael E Tipping. Analysis of sparse bayesian learning. *Advances in Neural Information Processing Systems*, 14:383–389, 2002.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. A note on the group lasso and a sparse group lasso. *arXiv preprint arXiv:1001.0736*, 2010.

Surya Ganguli and Haim Sompolinsky. Statistical mechanics of compressed sensing. *Physical review letters*, 104(18):188701, 2010.

Marcel V Gerven, Botond Cseke, Robert Oostenveld, and Tom Heskes. Bayesian source localization with the multivariate laplace prior. In *Advances in Neural Information Processing Systems*, pages 1901–1909, 2009.

Alexandre Gramfort, Bertrand Thirion, and Gaël Varoquaux. Identifying predictive regions from fmri with tv-l1 prior. In *Pattern Recognition in Neuroimaging (PRNI), 2013 International Workshop on*, pages 17–20. IEEE, 2013.

Logan Grosenick, Brad Klingenberg, Brian Knutson, and Jonathan E Taylor. A family of interpretable multivariate models for regression and classification of whole-brain fmri data. *arXiv preprint arXiv:1110.4139*, 2011.

Asela Gunawardana and William Byrne. Convergence theorems for generalized alternating minimization procedures. *The Journal of Machine Learning Research*, 6(Dec):2049–2073, 2005.

James V Haxby, M Ida Gobbini, Maura L Furey, Alumit Ishai, Jennifer L Schouten, and Pietro Pietrini. Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science*, 293(5539):2425–2430, 2001.

Daniel Hernández-Lobato and José Miguel Hernández-Lobato. Learning feature selection dependencies in multi-task learning. In *Advances in Neural Information Processing Systems*, pages 746–754, 2013.

Daniel Hernández-Lobato, José Miguel Hernández-Lobato, and Pierre Dupont. Generalized spike-and-slab priors for bayesian group feature selection using expectation propagation. *The Journal of Machine Learning Research*, 14(1):1891–1945, 2013.

Nicholas J Higham. *Accuracy and stability of numerical algorithms*. SIAM, 2002.

Junzhou Huang, Tong Zhang, and Dimitris Metaxas. Learning with structured sparsity. *The Journal of Machine Learning Research*, 12:3371–3412, 2011.

Laurent Jacob, Guillaume Obozinski, and Jean-Philippe Vert. Group lasso with overlap and graph lasso. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 433–440. ACM, 2009.

Rodolphe Jenatton, Jean-Yves Audibert, and Francis Bach. Structured variable selection with sparsity-inducing norms. *The Journal of Machine Learning Research*, 12:2777–2824, 2011.

Nancy Kanwisher, Josh McDermott, and Marvin M Chun. The fusiform face area: a module in human extrastriate cortex specialized for face perception. *The Journal of Neuroscience*, 17(11):4302–4311, 1997.

Seyoung Kim and Eric P Xing. Statistical estimation of correlated genome associations to a quantitative trait network. *PLoS genetics*, 5(8):e1000587, 2009.

Matthieu Kowalski, Kai Siedenburg, and Monika Dörfler. Social sparsity! neighborhood systems enrich structured shrinkage operators. *IEEE transactions on signal processing*, 61(10):2498–2511, 2013.

Miguel Lázaro-Gredilla, Joaquin Quiñonero-Candela, Carl Edward Rasmussen, and Aníbal R Figueiras-Vidal. Sparse spectrum gaussian process regression. *The Journal of Machine Learning Research*, 11:1865–1881, 2010.

Han Liu, Larry Wasserman, and John D Lafferty. Nonparametric regression and classification with joint sparsity constraints. In *Advances in Neural Information Processing Systems*, pages 969–976, 2009.

David JC MacKay. Bayesian interpolation. *Neural computation*, 4(3):415–447, 1992.

David JC MacKay. *Information theory, inference and learning algorithms.* Cambridge university press, 2003.

Daniel S Marcus, Tracy H Wang, Jamie Parker, John G Csernansky, John C Morris, and Randy L Buckner. Open access series of imaging studies (oasis): cross-sectional mri data in young, middle aged, nondemented, and demented older adults. *Journal of cognitive neuroscience*, 19(9):1498–1507, 2007.

Vincent Michel, Alexandre Gramfort, Gaël Varoquaux, Evelyn Eger, and Bertrand Thirion. Total variation regularization for fmri-based prediction of behavior. *Medical Imaging, IEEE Transactions on*, 30(7):1328–1340, 2011.

Toby J Mitchell and John J Beauchamp. Bayesian variable selection in linear regression. *Journal of the American Statistical Association*, 83(404):1023–1032, 1988.

Iain Murray and Ryan P Adams. Slice sampling covariance hyperparameters of latent gaussian models. In *Advances in Neural Information Processing Systems*, pages 1732–1740, 2010.

Iain Murray, Ryan Prescott Adams, and David JC MacKay. Elliptical slice sampling. *arXiv preprint arXiv:1001.0175*, 2009.

Radford M Neal. *Bayesian learning for neural networks.* PhD thesis, University of Toronto, 1995.

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

Trevor Park and George Casella. The bayesian lasso. *Journal of the American Statistical Association*, 103(482):681–686, 2008.

Maneesh Sahani and Jennifer F Linden. Evidence optimization techniques for estimating stimulus-response functions. *Advances in Neural Information Processing Systems*, pages 317–324, 2003.

Alexander Schmolck. *Smooth Relevance Vector Machines*. PhD thesis, University of Exeter, 2008.

Michael L Stein. *Interpolation of Spatial Data: Some Theory for Kriging*. Springer, 1999.

Wesley Tansey, Oluwasanmi Koyejo, Russell A. Poldrack, and James G. Scott. False discovery rate smoothing. *Journal of the American Statistical Association*, 0(ja):0–0, 2017. doi: 10.1080/01621459.2017.1319838.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

Michael E Tipping. Sparse bayesian learning and the relevance vector machine. *The Journal of Machine Learning Research*, 1:211–244, 2001.

Michael E Tipping and Anita C Faul. Fast marginal likelihood maximisation for sparse bayesian models. In C. M. Bishop and B. J. Frey, editors, *Proceedings of the ninth international workshop on artificial intelligence and statistics*, volume 1(3), pages 1–13. Citeseer, 2003.

Sabrina M Tom, Craig R Fox, Christopher Trepel, and Russell A Poldrack. The neural basis of loss aversion in decision-making under risk. *Science*, 315(5811):515–518, 2007.

Sabrina M Tom, Craig R Fox, Christopher Trepel, and Russell A Poldrack. Mixed-gambles task, 2011. This data was obtained from the OpenfMRI database. Its accession number is ds000005.

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.

Anqi Wu, Nicholas G Roy, Stephen Keeley, and Jonathan W Pillow. Gaussian process based nonlinear latent structure discovery in multivariate spike train data. In *Advances in Neural Information Processing Systems*, pages 3496–3505, 2017.

Lei Yu, Hong Sun, Jean-Pierre Barbot, and Gang Zheng. Bayesian compressive sensing for cluster structured sparse signals. *Signal Processing*, 92(1):259–269, 2012.

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

Willard I Zangwill. *Nonlinear programming: a unified approach.* Prentice-Hall, 1969.

Han Zhou and Brian Gaines. Matlab sparsereg toolbox version 1.0.0. `http://hua-zhou.github.io/SparseReg/`, 2017.

Appendix A. The Hessian of the negative log-posterior in Laplace approximation for smooth-DRD

Now we derive the Hessian matrix inside of the inverse of (eq. 22).

$$H = -\frac{\partial^2}{\partial \mathbf{u} \partial \mathbf{u}^\top} \Big[\log p(\mathbf{y}|X, \mathbf{u}, \sigma^2, \delta) + \log p(\mathbf{u}|\boldsymbol{\theta}_{drd}) \Big]. \tag{52}$$

The first term to take the partial derivatives of is (eq. 20):

$$\log p(\mathbf{y}|X, \mathbf{u}, \sigma^2, \delta) = -\frac{1}{2} \log |XCX^\top + \sigma^2 I| - \frac{1}{2} \mathbf{y}^\top (XCX^\top + \sigma^2 I)^{-1} \mathbf{y} + const, \tag{53}$$

and the second is (eq. 21):

$$\log p(\mathbf{u}|\boldsymbol{\theta}_{drd}) = -\frac{1}{2} (\mathbf{u} - b\mathbf{1})^\top K^{-1} (\mathbf{u} - b\mathbf{1}) - \frac{1}{2} \log |K| + const. \tag{54}$$

Define $S = XCX^\top + \sigma^2 I$, where $C = C_{drd}^{\frac{1}{2}} \Sigma C_{drd}^{\frac{1}{2}}$. Let $Z = X C_{drd}^{\frac{1}{2}} \Sigma^{\frac{1}{2}}$, then,

$$\begin{aligned} S &= X C_{drd}^{\frac{1}{2}} \Sigma C_{drd}^{\frac{1}{2}} X^\top + \sigma^2 I = ZZ^\top + \sigma^2 I, & (55) \\ \log p(\mathbf{u}|\mathbf{y}, X, \boldsymbol{\theta}) &= -\frac{1}{2} \mathbf{y}^\top S^{-1} \mathbf{y} - \frac{1}{2} \log |S| - \frac{1}{2} (\mathbf{u} - b\mathbf{1})^\top K^{-1} (\mathbf{u} - b\mathbf{1}) + const. & (56) \end{aligned}$$

The first derivative with respect to $\mathbf{u}_i$ is given by:

$$\begin{aligned} \frac{\partial}{\partial \mathbf{u}_i} \log p(\mathbf{u}|\mathbf{y}, X, \boldsymbol{\theta}) &= \frac{1}{2} \frac{\partial}{\partial \mathbf{u}_i} \Big(-\mathbf{y}^\top S^{-1} \mathbf{y} - \log |S| - (\mathbf{u} - b\mathbf{1})^\top K^{-1} (\mathbf{u} - b\mathbf{1}) \Big) & (57) \\ &= \mathrm{Tr} \left[Z^\top S^{-1} \mathbf{y} \mathbf{y}^\top S^{-1} \left(\frac{\partial}{\partial \mathbf{u}_i} Z \right) - Z^\top S^{-1} \left(\frac{\partial}{\partial \mathbf{u}_i} Z \right) \right] & (58) \\ &\quad - \left[K^{-1} (\mathbf{u} - b\mathbf{1}) \right]_i , & (59) \end{aligned}$$

where

$$\frac{\partial}{\partial \mathbf{u}_i} Z = X \left(\frac{\partial}{\partial \mathbf{u}_i} C_{drd}^{\frac{1}{2}} \right) \Sigma^{\frac{1}{2}}. \tag{60}$$

The second derivative with respect to $\mathbf{u}_j$ is given by:

$$\frac{\partial^2}{\partial \mathbf{u}_i \partial \mathbf{u}_j} \log p(\mathbf{u}|\mathbf{y}, X, \boldsymbol{\theta}) = H_1 + H_2 + H_3 - K_{ij}^{-1} = -H_{ij}, \tag{61}$$

$$H_1 = \mathrm{Tr} \left[Z^\top S^{-1} \mathbf{y} \mathbf{y}^\top S^{-1} \left(\frac{\partial^2}{\partial \mathbf{u}_i \partial \mathbf{u}_j} Z \right) - Z^\top S^{-1} \left(\frac{\partial^2}{\partial \mathbf{u}_i \partial \mathbf{u}_j} Z \right) \right], \tag{62}$$

$$H_2 = \mathrm{Tr} \left[\left(\frac{\partial}{\partial \mathbf{u}_j} Z \right)^\top S^{-1} \mathbf{y} \mathbf{y}^\top S^{-1} \left(\frac{\partial}{\partial \mathbf{u}_i} Z \right) - \left(\frac{\partial}{\partial \mathbf{u}_j} Z \right)^\top S^{-1} \left(\frac{\partial}{\partial \mathbf{u}_i} Z \right) \right], \tag{63}$$

$$H_3 = 2\mathrm{Tr}\left[-Z^\top S^{-1}\left(\frac{\partial}{\partial \mathbf{u}_j}Z\right)Z^\top S^{-1}\mathbf{y}\mathbf{y}^\top S^{-1}\left(\frac{\partial}{\partial \mathbf{u}_i}Z\right)\right. \quad (64)$$

$$-Z^\top S^{-1}\mathbf{y}\mathbf{y}^\top S^{-1}\left(\frac{\partial}{\partial \mathbf{u}_j}Z\right)Z^\top S^{-1}\left(\frac{\partial}{\partial \mathbf{u}_i}Z\right) \quad (65)$$

$$\left.+Z^\top S^{-1}\left(\frac{\partial}{\partial \mathbf{u}_j}Z\right)Z^\top S^{-1}\left(\frac{\partial}{\partial \mathbf{u}_i}Z\right)\right], \quad (66)$$

where

$$\frac{\partial^2}{\partial \mathbf{u}_i \partial \mathbf{u}_j}Z = X\left(\frac{\partial^2}{\partial \mathbf{u}_i \partial \mathbf{u}_j}C_{drd}^{\frac{1}{2}}\right)\Sigma^{\frac{1}{2}}. \quad (67)$$

For DRD only, we can just derive the Hessian by replacing $C_{drd}^{\frac{1}{2}}\Sigma^{\frac{1}{2}}$ with $C_{drd}^{\frac{1}{2}}$.

Appendix B. Proof of convexity of $\mathcal{L}_1(\mathbf{u})$

We ignore the scaling $\frac{1}{2}$ here for simplicity, and write,

$$\begin{aligned}\mathcal{L}_1(\mathbf{u}) &= \log|X\mathrm{diag}(e^{\mathbf{u}})X^\top + \sigma^2 I| & (68)\\ &= \log|\mathrm{diag}(e^{\mathbf{u}})\frac{X^\top X}{\sigma^2} + I| + const & (69)\\ &= \log|\frac{X^\top X}{\sigma^2} + \mathrm{diag}(e^{-\mathbf{u}})| + \log|\mathrm{diag}(e^{\mathbf{u}})| + const. & (70)\end{aligned}$$

Let $V = \frac{X^\top X}{\sigma^2}$, which is p.s.d., then

$$\begin{aligned}&\frac{\partial}{\partial \mathbf{u}}\log|V + \mathrm{diag}(e^{-\mathbf{u}})| = -\mathrm{diag}\left((\mathrm{diag}(e^{-\mathbf{u}}) + V)^{-1} \odot \mathrm{diag}(e^{-\mathbf{u}})\right)\\ &\frac{\partial^2}{\partial^2 \mathbf{u}}\log|V + \mathrm{diag}(e^{-\mathbf{u}})| = \left(V(\mathrm{diag}(e^{\mathbf{u}})V + I)^{-1}\right) \odot \left(\mathrm{diag}(e^{-\mathbf{u}}) + V\right)^{-1},\end{aligned} \quad (71)$$

where $\odot$ is the Hadamard product. Moreover, we know that

$$\begin{aligned}V(\mathrm{diag}(e^{\mathbf{u}})V + I)^{-1} &= \frac{X^\top X}{\sigma^2}(\mathrm{diag}(e^{\mathbf{u}})\frac{X^\top X}{\sigma^2} + I)^{-1}\\ &= X^\top(X\mathrm{diag}(e^{\mathbf{u}})X^\top + \sigma^2 I)^{-1}X \succeq 0.\end{aligned} \quad (72)$$

Thanks to the Schur product theorem stating that the Hadamard product of two positive semi-definite matrices is also a positive semi-definite matrix, we have $\frac{\partial^2}{\partial^2 \mathbf{u}}\log|V + \mathrm{diag}(e^{-\mathbf{u}})| \succeq 0$, thus $\log|\frac{X^\top X}{\sigma^2} + \mathrm{diag}(e^{-\mathbf{u}})|$ is convex in $\mathbf{u}$. In addition, $\log|\mathrm{diag}(e^{\mathbf{u}})|$ is also convex in $\mathbf{u}$. Therefore, $\mathcal{L}_1(\mathbf{u})$ is convex in $\mathbf{u}$.

Appendix C. Proof of boundedness and non-emptiness of $\mathcal{F}(\mathbf{u})$

We want to prove that for (eq. 35), when $||\mathbf{u}|| \to \infty$, we have $\mathcal{F}(\mathbf{u}) \to \infty$. Rewrite $\mathcal{F}$ here,

$$\mathcal{F}(\mathbf{u}) = \mathbf{z}^{k\top}\mathbf{h}(\mathbf{u}) - \mathcal{L}^*_{\mathbf{h}}(\mathbf{z}) + \mathcal{L}_1(\mathbf{u}) + \mathcal{L}_3(\mathbf{u}), \quad (73)$$

where $\mathbf{h}(\mathbf{u}) = e^{-\mathbf{u}}$, $\mathcal{L}_1(\mathbf{u}) = \frac{1}{2}\log|X\mathrm{diag}(e^{\mathbf{u}})X^\top + \sigma^2 I|$ and $\mathcal{L}_3(\mathbf{u}) = \frac{1}{2}(\mathbf{u}-b\mathbf{1})^\top K^{-1}(\mathbf{u}-b\mathbf{1})$.

1) Each element in $\mathbf{h}(\mathbf{u})$ is bounded by 0 and 1. Thus when $||\mathbf{u}|| \to \infty$, ${\mathbf{z}^k}^\top \mathbf{h}(\mathbf{u})$ will be bounded.
2) K^{-1} is a positive semi-definite (p.s.d.) matrix. Thus $\mathcal{L}_3(\mathbf{u})$ is lower-bounded by 0. When $||\mathbf{u}|| \to \infty$, $\mathcal{L}_3(\mathbf{u}) \geq 0$. The upper bound is unclear.
3) Denote $\Gamma = \mathrm{diag}(e^{\mathbf{u}})$ whose diagonal values are all nonnegative. Now we want to prove when $||\mathbf{u}|| \to \infty$, we have $\mathcal{L}_1(\mathbf{u}) = \frac{1}{2}\log|X\Gamma X^\top + \sigma^2 I| \to \infty$.

First, we note that $\mathcal{L}_1(\mathbf{u})$ has a lower bound. It can be easily shown that the eigenvalues of $X\Gamma X^\top + \sigma^2 I$ should be greater than or equal to σ^2, given $X\Gamma X^\top$ is a p.s.d. matrix.

If $||\mathbf{u}|| \to \infty$, we can assume $u_1 \to \infty, ..., u_s \to \infty$ where $\mathbf{u} \in \mathbb{R}^p$ and $s \leq p$, then $e^{u_i} \to \infty$, for all $i \in \{1, ..., s\}$. For $\{u_i\}_{i=s+1}^p$, if u_i is finite, e^{u_i} will be finite; else if $u_i \to -\infty$, $e^{u_i} = 0$. Thus e^{u_i} is a finite value for all $i \in \{s+1, ..., p\}$. We can write Γ as an addition of two matrices A and B, i.e. $\Gamma = A + B$.

$$A = \begin{bmatrix} e^{u_1} & & & & & \\ & e^{u_2} & & & 0 & \\ & & \ddots & & & \\ & & & e^{u_s} & & \\ & & & & 0 & \\ & 0 & & & & \ddots & \\ & & & & & & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & & & & & \\ & 0 & & & 0 & \\ & & \ddots & & & \\ & & & 0 & & \\ & & & & e^{u_{s+1}} & \\ & 0 & & & & \ddots & \\ & & & & & & e^{u_p} \end{bmatrix} \tag{74}$$

The nonzero elements in A are infinite values. The nonzero elements in B are finite nonnegative values. Let $M = XBX^\top + \sigma^2 I \in \mathbb{R}^{n\times n}$. $XBX^\top$ is a p.s.d. matrix. The smallest eigenvalue of $XBX^\top$ should be nonnegative. Therefore the smallest eigenvalue of M is greater than or equal to σ^2. This implies the invertibility of M. Since M^{-1} is also positive definite, we can factorize M^{-1} into $S \in \mathbb{R}^{n\times n}$ and $S^\top$, i.e. $M^{-1} = SS^\top$. Thus, we can write

$$\begin{aligned} \mathcal{L}_1(\mathbf{u}) &= \frac{1}{2}\log|XAX^\top + XBX^\top + \sigma^2 I| && (75) \\ &= \frac{1}{2}\log|XAX^\top + M| && (76) \\ &= \frac{1}{2}\log|XAX^\top M^{-1} + I| + \frac{1}{2}\log|M| && (77) \\ &= \frac{1}{2}\log|XAX^\top SS^\top + I| + \frac{1}{2}\log|M| && (78) \\ &= \frac{1}{2}\log|S^\top XAX^\top S + I| + \frac{1}{2}\log|M| && (79) \end{aligned}$$

Combining $X^\top S$ to be one matrix $Z \in \mathbb{R}^{p\times n}$, we can investigate the elements in $Z^\top AZ$. $Z^\top AZ$ should be equal to $\infty * \widetilde{Z}^\top \widetilde{Z}$ where $\widetilde{Z} \in \mathbb{R}^{s\times n}$ is a trimmed Z by throwing away the rows with indices from $s+1$ to p. Therefore, nonzero eigenvalues of $\widetilde{Z}^\top \widetilde{Z}$ will turn into ∞ in $Z^\top AZ$. Zero eigenvalues will remain zero.

Let $\lambda_i \geq 0$ denote the ith eigenvalue of $Z^\top AZ$, then

$$\mathcal{L}_1(\mathbf{u}) \quad = \quad \frac{1}{2}\sum_{i=1}^{p}\log(\lambda_i + 1) + \frac{1}{2}\log|M| \tag{80}$$

Since there exists at least one eigenvalue λ_i in $Z^\top AZ$ approaching ∞, we can conclude that $\mathcal{L}_1(\mathbf{u})$ also approaches ∞ in such a case.

Accordingly, if $||\mathbf{u}|| \to \infty$, we have $\mathcal{F}(\mathbf{u}) \to \infty$, then there must exist a solution set for minimizing $\mathcal{F}(\mathbf{u})$. This validates the nonemptiness of the solution set. Furthermore, the solution set must be bounded. If it's not bounded, there must be a solution at ∞ with the minimal $\mathcal{F}(\mathbf{u})$, but this contradicts the assumption that $\mathcal{F}(\mathbf{u}) \to \infty$ when $||\mathbf{u}|| \to \infty$. Therefore, we can claim that the solution set of $\mathcal{F}(\mathbf{u})$ is bounded and nonempty.