THE GROUP-LASSO: TWO NOVEL APPLICATIONS

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF STATISTICS
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Michael Lim
August 2013

Preface

This thesis deals with two problems: learning linear interaction models, and electroencephalography (EEG) source estimation in the visual cortex. These are quite different problems, but they have a common theme that brings them together: we propose solutions to each that are based on the group-lasso. The group-lasso was first introduced in [49], and is an example of regularization applied to a supervised learning problem.

Learning interactions is a diffcult problem because of the number of variables involved. With a million variables (as in the case of genome wide association studies (GWAS), we are already looking at a candidate interaction search space of about a trillion terms. Even if the computational problems are overcome, there is still the statistical issue of spurious correlations and low signal to noise ratios inherent in these problems. Past approaches to this problem such as hierNet [5] and logic regression [33] are limited in either their computational feasibility or the types of variables they can accommodate. Our contribution here is GLINTERNET, a method for learning pairwise interactions via hierarchical group-lasso regularization. We demonstrate that GLINTERNET is competitive with current methods (where these methods are feasible), but also that it has the added advantage of being able to accommodate both categorical variables with arbitrary numbers of levels as well as continuous variables. GLINTERNET is available as a package on CRAN for the statistical software R.

Our EEG source problem arises from visual activation studies. Here, subjects are given visual stimuli, and their neural response is measured in a non-invasive manner through the use the sensors (or channels) placed around the subject's head. The goal is to recover the underlying neural activity (the sources) that are responsible for the observations recorded with these sensors. One method that is currently used to perform source inversion is called the minimum norm, which applies ridge regression to the observed sensor readings. We make two contributions in this domain. First, we show that the group-lasso outperforms the minimum norm inversion, and that the group-lasso performance improves with the number of subjects. This occurs because the group-lasso is able to pool information across multiple subjects, whereas the minimum norm is inherently unable to do so. A post-processing step may be applied to the minimum norm estimates by averaging across multiple subjects. Our second contribution consists of showing that averaging within appropriately defined regions of interest (ROIs) in the visual cortex across multiple subjects is able to dramatically boost the performance of both the minimum norm and group-lasso solutions, and also improves with the number of subjects. These two contributions, to the best of our knowledge, are novel results.

There are four chapters in this thesis. Chapter 1 introduces the group-lasso and describes some of its properties. Chapter 2 consists of the interaction learning problem. We make clear the problem statement, make the case for hierarchical interaction models, and then present our solution. The EEG source estimation problem is tackled in Chapter 3. We introduce the problem, discuss past approaches and why they are inadequate before presenting our solution that is based on the group-lasso. Finally, we conclude with a discussion in Chapter 4.

Acknowledgements

I would like to thank my advisor, Trevor "The Major" Hastie, for his guidance over the past 3 years. Prior to starting my studies at Stanford, I was often advised to place the utmost importance on picking an advisor, because the advisor can make or break your experience. I confess I did not heed any of that, but if I had to go back and rechoose, it would still have to be Trevor. He has not just been an academic advisor but also a mentor and friend.

I also thank my committee members Rob, Jonathan, Brad, and Tony for their willingness to listen, advise, and taking the time to be on my committee.

Thanks also go out to my collaborators Tony, Justin, and Benoit, without whom the EEG work would not have been possible. The collaboration gave me the opportunity to learn about EEG signal estimation, a field I would likely have never come across otherwise.

Thanks to my friends both within and outside the statistics department. Our discussions have often impacted my approach to thinking about problems, and have helped a great deal toward the completion of this work. Special thanks to CS and Noah for computational and programming discussions. And thanks to CS, Winston, Alexandra, and Tian Tsong for pre-release glinternet testing. Rahul was also instrumental in optimization pointers.

Special thanks to my fiancee Flora, who has waited (im)patiently for the past four years for me to finish up. Without her support and motivation, I could not have graduated.

Finally, I thank my parents who have dedicated their lives to raising me. I dedicate this work to them.

Contents

List of Tables

List of Figures

Chapter 1

Introduction and preliminaries

In this chapter, we introduce the lasso and the group-lasso. The group-lasso can be viewed as an extension of the lasso to groups of variables, and is the common theme running throughout the thesis.

1.1 The lasso

Regularization plays an important role in statistics and machine learning. One example arises in the context of linear models applied to large datasets. Advances in technology has made it possible to store ever larger amounts of data, and while the number of observations has increased dramatically, so too has the number of features. In fact, the number of features often exceeds the number of data points; this is commonly referred to as the "$p > n$" problem. The linear regression problem is ill-posed in this situation because there is no unique solution: there are infinitely many coefficient vectors that give the same fit.

A popular approach in supervised learning problems of this type is to use regularization, such as adding a squared L_2 penalty of the form $\|\beta\|_2^2$ or a L_1 penalty of the form $\|\beta\|_1$ to the model coefficients. The latter type of penalty has been the focus of much research since its introduction in [40], and is called the lasso. The lasso obtains the estimates $\hat{\beta}$ as the solution to

$$\text{argmin}_{\mu,\beta}\,\mathcal{L}(\mathbf{Y},\mathbf{X};\mu,\beta)+\lambda\|\beta\|_1, \tag{1.1}$$

where $\mathbf{Y}$ is a vector of observations, $\mathbf{X}$ is the feature matrix (or design matrix), μ is the intercept, and β is the vector of coefficients to be estimated. $\mathcal{L}$ can be any loss function, most commonly squared error loss

$$\mathcal{L}(\mathbf{Y},\mathbf{X};\mu,\beta)=\frac{1}{2}\left\|\mathbf{Y}-\mu\cdot\mathbf{1}-\mathbf{X}\beta\right\|_2^2 \tag{1.2}$$

if the observations $\mathbf{Y}$ are quantitative, or logistic loss

$$\mathcal{L}(\mathbf{Y},\mathbf{X};\mu,\beta)=-\left[\mathbf{Y}^T(\mu\cdot\mathbf{1}+\mathbf{X}\beta)-\mathbf{1}^T\log\left(\mathbf{1}+\exp(\mu\cdot\mathbf{1}+\mathbf{X}\beta)\right)\right] \tag{1.3}$$

if the observations are binary (log and exp taken component wise).

The L_1 penalty has the effect of performing variable selection by setting some of the coefficients to zero. The parameter λ controls the amount of regularization, and for large enough values, all the coefficients will be estimated to be zero.

1.2 The group-lasso

There is a group analog to the lasso, called the group-lasso [49], that sets groups of coefficients to zero. Suppose there are p groups of variables (possibly of different sizes), and let the feature matrix for group i be denoted by $\mathbf{X}_i$. Let $\mathbf{Y}$ denote the vector of observations. The group-lasso obtains the estimates $\hat{\beta}_j$, $j=1,\ldots,p$, as the solution to

$$\text{argmin}_{\mu,\beta}\,\frac{1}{2}\|\mathbf{Y}-\mu\cdot\mathbf{1}-\sum_{j=1}^{p}\mathbf{X}_j\beta_j\|_2^2+\lambda\sum_{j=1}^{p}\gamma_j\|\beta_j\|_2. \tag{1.4}$$

Note that the L_2 penalty in (1.4) is not squared, and that if each group consists of only one variable, this reduces to the lasso criterion in (1.1). Just as the lasso performs variable selection by estimating some of the coefficients to be zero, the group-lasso does selection on the group level: it is able to zero out *groups* of coefficients. If an estimate $\hat{\beta}_i$ is nonzero, then *all* its components are usually nonzero.

The parameter λ controls the amount of regularization, with larger values implying more regularization. When λ is large enough, all the coefficients will be estimated as zero. The γ's allow each group to be penalized to different extents, which allows us to penalize some groups more (or less) than others. To solve (1.4), we start with λ large enough so that all estimates are zero. Decreasing λ along a grid of values results in a path of solutions, from which an optimal λ can be chosen by cross validation or some model selection procedure. In particular, for the EEG source estimation problem, we use generalized cross validation (GCV) [14] with a heuristic for the degrees of freedom using the results in [22].

The Karush-Kuhn-Tucker (KKT) optimality conditions for the group-lasso are simple to compute and check. For group i, they are

$$\|\mathbf{X}_i^T(\mathbf{Y} - \hat{\mathbf{Y}})\|_2 < \gamma_i\lambda \quad \text{if} \quad \hat{\beta}_i = 0 \tag{1.5}$$

$$\|\mathbf{X}_i^T(\mathbf{Y} - \hat{\mathbf{Y}})\|_2 = \gamma_i\lambda \quad \text{if} \quad \hat{\beta}_i \neq 0. \tag{1.6}$$

where $\hat{\mathbf{Y}} = \hat{\mu}\cdot\mathbf{1} + \sum_{i=1}^p \mathbf{X}_i\hat{\beta}$ is the vector of fitted values. The group-lasso is commonly fit with some form of gradient descent, and convergence can be confirmed by checking that the solutions satisfy the KKT conditions. We use an adaptive version of fast iterative soft thresholding (FISTA) [3] and cyclic group-wise coordinate descent in our applications. See Sections 2.8 and 3.5 for more details.

Adding a ridge penalty to (1.4) results in the group analog of the elastic-net:

$$\text{argmin}_{\mu,\beta}\,\frac{1}{2}\left\|\mathbf{Y} - \mu\cdot\mathbf{1} - \sum_{j=1}^p \mathbf{X}_j\beta_j\right\|_2^2 + \lambda\sum_{j=1}^p \gamma_j\|\beta_j\|_2 + \alpha\|\beta\|_2^2. \tag{1.7}$$

The squared L_2 penalty in (1.7) applies to the entire coefficient vector β. This allows us to reduce the variance in the estimates $\hat{\beta}_i$, and having $\alpha > 0$ is helpful in our EEG experiments. Now that we have two parameters λ and α, a two-dimensional grid search has to be done to select optimal values for them. Since this can be computationally intensive, we keep α fixed and do the grid search only on λ. We discuss how we select α in Appendix A.

The overlapped group-lasso is a variant of the group-lasso where the groups of variables are allowed to have overlaps, i.e. some variables can show up in more than one group. However, each time a variable shows up in a group, it gets a new coefficient. For example, if a variable is included in 3 groups, then it has 3 coefficients that need to be estimated. We refer the reader to [21] for more details.

1.2.1 Extending the group-lasso to matrix-valued coefficients

Our description of the group-lasso above treated the coefficients β as a vector. Since the EEG source activity at a single time point is a vector, and we wish to recover the activity over several time points, we need to be able to handle the case where the coefficients are matrices. This can be done via a straightforward extension of (1.7). As before let $\mathbf{X}_i$ denote the feature matrix for group i and let $\mathbf{Y}$ be the $N \times T$ *matrix* of observations.

$$\operatorname{argmin}_{\mu,\beta} \frac{1}{2} \left\| \mathbf{Y} - \mathbf{1}\mu^t - \sum_{j=1}^{p} \mathbf{X}_j \beta_j \right\|_F^2 + \lambda \sum_{j=1}^{p} \gamma_j \|\beta_j\|_F + \alpha \|\beta\|_F^2. \tag{1.8}$$

We now have T intercepts (one for each column of $\mathbf{Y}$), and we use the Frobenius norm $\|\cdot\|_F$ instead of the L_2 norm. The solutions have the same property as before in that if $\hat{\beta}_i$ is nonzero, then *all* its components are usually nonzero.

Chapter 2

Learning interactions

Our first application of the group-lasso is in the context of learning linear pairwise interaction models via hierarchical group-lasso regularization. We begin by defining what an interaction means and describing the difficulty of the problem.

2.1 Introduction

Given an observed response and explanatory variables, we expect interactions to be present if the response cannot be explained by additive functions of the variables. The following definition makes this more precise.

Definition 1. *When a function $f(x, y)$ cannot be expressed as $g(x) + h(y)$ for some functions g and h, we say that there is an interaction in f between x and y.*

Interactions of single nucleotide polymorphisms (SNPs) are thought to play a role in cancer [36] and other diseases. Modeling interactions has also served the recommender systems community well: latent factor models (matrix factorization) aim to capture user-item interactions that measure a user's affinity for a particular item, and are the state of the art in predictive power [23]. In lookalike-selection, a problem that is of interest in computational advertising, one looks for features that most separates a group of observations from its complement, and it is conceivable that interactions among the features can play an important role.

There are many challenges, the first of which is a problem of scalability. Even with 10,000 variables, we are already looking at a 50×10^6-dimensional space of possible interaction pairs. Complicating the matter are correlations amongst the variables, which makes learning even harder. Finally, in some applications, sample sizes are relatively small and the signal to noise ratio is low. For example, genome wide association studies (GWAS) can involve hundreds of thousands or millions of variables, but only several thousand observations. Since the number of interactions is on the order of the square of the number of variables, computational considerations quickly become an issue.

Finding interactions is an example of the "$p > n$" problem where there are more features or variables than observations. This is a natural setting for regularization, and the idea behind our method is to set up main effects and interactions (to be defined later) as a group of variables, and then we perform selection via the group-lasso.

Discovering interactions is an area of active research; see, for example, [5] and [7]. In this thesis, we introduce GLINTERNET, a method for learning first-order interactions that can be applied to categorical variables with arbitrary numbers of levels, continuous variables, and combinations of the two. Our approach consists of two phases: a screening stage (for large problems) that gives a candidate set of main effects and interactions, followed by variable selection on the candidate set with the group-lasso. We introduce two screening procedures, the first of which is inspired by our observation that boosting with depth-2 trees naturally gives rise to an interaction selection process that enforces hierarchy: an interaction cannot be chosen until a split has been made on one of its two associated main effects. The second method is an adaptive procedure that is based on the strong rules [41] for discarding predictors in lasso-type problems. We show in Section 2.3 how the group-lasso penalty naturally enforces strong hierarchy in the resulting solutions.

We can now give an overview of our method:

1. If required, screen the variables to get a candidate set $\mathcal{C}$ of interactions and their associated main effects. Otherwise, take $\mathcal{C}$ to consist of all main effects and pairwise interactions.

2. Fit a group-lasso on $\mathcal{C}$ with a grid of values for the regularization parameter. Start with $\lambda = \lambda_{max}$ for which all estimates are zero. As we decrease λ, we allow more terms to enter the model, and we stop once a user-specified number of interactions have been discovered. Alternatively, we can choose λ using any model selection technique such as cross validation.

2.1.1 A simulated example

As a first example, we perform 100 simulations with 500 3-level categorical variables and 800 quantitative observations. There are 10 main effects and 10 interactions in the ground truth, and the noise level is chosen to give a signal to noise ratio of one. We run GLINTERNET without any screening, and stop after ten interactions have been found. The average false discovery rate and standard errors are plotted as a function of the number of interactions found in Figure 2.1.

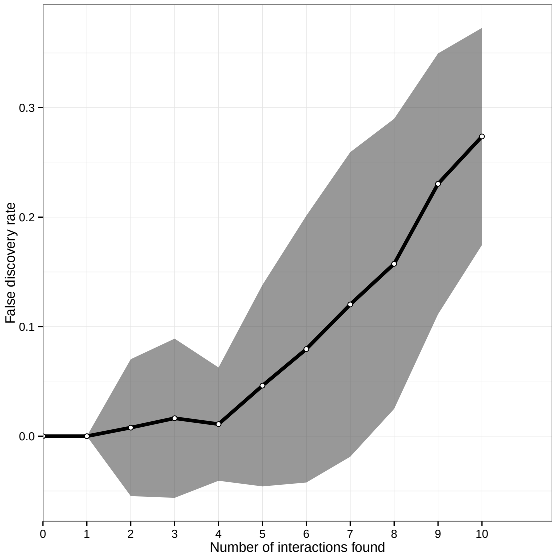


Figure 2.1: False discovery rate vs number of discovered interactions

2.1.2 Organization of Chapter 2

The rest of the chapter is organized as follows. Section 2.2 introduces basic notions and notation. In section 2.3, we introduce the group-lasso and how it fits into our framework for finding interactions. We also show how GLINTERNET is equivalent to an overlapping grouped lasso. We discuss screening in Section 2.4, and give several examples with both synthetic and real datasets in Section 2.7 before going into algorithmic details in Section 2.8. We conclude with a discussion in Section 2.9.

2.2 Background and notation

We use the random variables Y to denote the observed response, F to denote a categorical feature, and Z to denote a continuous feature. We use L to denote the number of levels that F can take. For simplicity of notation we will use the first L positive integers to represent these L levels, so that F takes values in the set $\{i \in \mathbb{Z} : 1 \leq i \leq L\}$. Each categorical variable has an associated random variable $X \in \mathbb{R}^L$ with a 1 that indicates which level F takes, and 0 everywhere else.

When there are p categorical (or continuous) features, we will use subscripts to index them, i.e. $F_1, \ldots, F_p$. Boldface font will always be reserved for vectors or matrices that comprise of realizations of these random variables. For example, $\mathbf{Y}$ is the n-vector of observations of the random variable Y, $\mathbf{F}$ is the n-vector of realizations of the random variable F, and $\mathbf{Z}$ is the n-vector of realizations of the random variable Z. Similarly, $\mathbf{X}$ is a $n \times L$ indicator matrix whose i-th row consists of a 1 in the $\mathbf{F}_i$-th column and 0 everywhere else. We use a $n \times (L_i \cdot L_j)$ indicator matrix $\mathbf{X}_{i:j}$ to represent the interaction $F_i : F_j$. We will write

$$\mathbf{X}_{i:j} = \mathbf{X}_i * \mathbf{X}_j, \tag{2.1}$$

where the first L_j columns of $\mathbf{X}_{i:j}$ are obtained by taking the elementwise products between the first column of $\mathbf{X}_i$ and the columns of $\mathbf{X}_j$, and likewise for the other columns. For example,

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} * \begin{pmatrix} e & f \\ g & h \end{pmatrix} = \begin{pmatrix} ae & af & be & bf \\ cg & ch & dg & dh \end{pmatrix}. \tag{2.2}$$

2.2.1 Definition of interaction for categorical variables

To see how Definition 1 applies to this setting, let $\mathbb{E}(Y|F_1 = i, F_2 = j) = \mu_{ij}$, the conditional mean of Y given that F_1 takes level i, and F_2 takes level j. There are 4 possible cases:

1. $\mu_{ij} = \mu$ (no main effects, no interactions)
2. $\mu_{ij} = \mu + \theta_1^i$ (one main effect F_1)
3. $\mu_{ij} = \mu + \theta_1^i + \theta_2^j$ (two main effects)
4. $\mu_{ij} = \mu + \theta_1^i + \theta_2^j + \theta_{1:2}^{ij}$ (main effects and interaction)

Note that all but the first case is overparametrized, and the usual procedure is to impose sum constraints on the main effects and interactions:

$$\sum_{i=1}^{L_1} \theta_1^i = 0, \quad \sum_{j=1}^{L_2} \theta_2^j = 0 \tag{2.3}$$

and

$$\sum_{i=1}^{L_1} \theta_{1:2}^{ij} = 0 \text{ for fixed } j, \quad \sum_{j=1}^{L_2} \theta_{1:2}^{ij} = 0 \text{ for fixed } i. \tag{2.4}$$

In what follows, θ_i, $i = 1, \cdots, p$, will represent the main effect coefficients, and $\theta_{i:j}$ will denote the interaction coefficients. We will use the terms "main effect coefficients" and "main effects" interchangeably, and likewise for interactions.

2.2.2 Weak and strong hierarchy

An interaction model is said to obey strong hierarchy if an interaction can be present only if both of its main effects are present. Weak hierarchy is obeyed as long as either of its main effects are present. Since main effects as defined above can be viewed as deviations from the global mean, and interactions are deviations from the main effects, it rarely make sense to have interactions without main effects. This leads us to prefer interaction models that are hierarchical. We will see in Section 2.3 that GLINTERNET produces estimates that obey strong hierarchy.

2.2.3 First order interaction model

Our model for a quantitative response Y is given by

$$Y = \mu + \sum_{i=1}^{p} X_i\theta_i + \sum_{i<j} X_{i:j}\theta_{i:j} + \epsilon \tag{2.5}$$

where $\epsilon \sim N(0, \sigma^2)$. For binary responses, we have

$$logit(\mathbb{P}(Y = 1|X)) = \mu + \sum_{i=1}^{p} X_i\theta_i + \sum_{i<j} X_{i:j}\theta_{i:j}. \tag{2.6}$$

We fit these models by minimizing an appropriate choice of loss function $\mathcal{L}$. Because the models are still overparametrized, we impose the relevant constraints for the coefficients θ (see (2.3) and (2.4)). We can thus cast the problem of fitting a first-order interaction model as an optimization problem with constraints:

$$\text{argmin}_{\mu,\theta} \mathcal{L}(\mathbf{Y}, \mathbf{X}_{i:i\leq p}, \mathbf{X}_{i:j}; \mu, \theta) \tag{2.7}$$

subject to the relevant constraints. $\mathcal{L}$ can be any loss function, typically squared error loss for the quantitative response model given by

$$\mathcal{L}(\mathbf{Y}, \mathbf{X}_{i:i\leq p}, \mathbf{X}_{i:j}; \mu, \theta) = \frac{1}{2}\left\| \mathbf{Y} - \mu \cdot \mathbf{1} - \sum_{i=1}^{p} \mathbf{X}_i\theta_i + \sum_{i<j} \mathbf{X}_{i:j}\theta_{i:j} \right\|_2^2, \tag{2.8}$$

and logistic loss for the binomial response model given by

$$\begin{aligned}\mathcal{L}(\mathbf{Y}, \mathbf{X}_{i:i\leq p}, \mathbf{X}_{i:j}; \mu, \theta) = -\Bigg[&\mathbf{Y}^T(\mu \cdot \mathbf{1} + \sum_{i=1}^{p} \mathbf{X}_i \theta_i + \sum_{i<j} \mathbf{X}_{i:j}\theta_{i:j}) \\ &- \mathbf{1}^T \log\left(\mathbf{1} + \exp(\mu \cdot \mathbf{1} + \sum_{i=1}^{p} \mathbf{X}_i \theta_i + \sum_{i<j} \mathbf{X}_{i:j}\theta_{i:j})\right)\Bigg], \quad (2.9)\end{aligned}$$

where the log and exp are taken component-wise.

Because the coefficients in (2.7) are unpenalized, the solutions $\hat{\theta}_i$ satisfy strong hierarchy (they are usually *all* nonzero). If $p + \begin{pmatrix} p \\ 2 \end{pmatrix} > n$, this problem is ill-posed, resulting in infinitely many solutions. Adding a ridge penalty is one way to tackle this problem, and the solutions will also satisfy strong hierarchy. The question then arises as to how to fit interaction models whose solutions are sparse (variable selection effect) and also satisfy hierarchy. A lasso penalty will achieve sparsity, but there is no guarantee that the solutions will have any form of hierarchy. This is the goal of this work.

2.3 Methodology and results

We want to fit the first order interaction model in a way that obeys strong hierarchy. We show in Section 2.3.1 how this can be achieved by adding an overlapped group-lasso penalty to the objective in (2.7). We then show how this constrained overlapped group-lasso problem can be conveniently solved via an *unconstrained* group-lasso (without overlaps).

2.3.1 Strong hierarchy through overlapped group-lasso

Adding an overlapped group-lasso penalty to (2.7) is one way of obtaining solutions that satisfy the strong hierarchy property. The results that follow hold for both squared error and logistic loss, but we focus on the former for clarity.

Consider the case where there are two categorical variables F_1 and F_2 with L_1 and

L_2 levels respectively. Their indicator matrices are given by $\mathbf{X}_1$ and $\mathbf{X}_2$. We solve

$$\begin{aligned}\operatorname{argmin}_{\mu,\alpha,\tilde{\alpha}} \frac{1}{2}\left\| \mathbf{Y} - \mu\cdot\mathbf{1} - \mathbf{X}_1\alpha_1 - \mathbf{X}_2\alpha_2 - [\mathbf{X}_1\ \mathbf{X}_2\ \mathbf{X}_{1:2}]\begin{bmatrix}\tilde{\alpha}_1\\ \tilde{\alpha}_2\\ \alpha_{1:2}\end{bmatrix}\right\|_2^2 \\ + \lambda\left(\|\alpha_1\|_2 + \|\alpha_2\|_2 + \sqrt{L_2\|\tilde{\alpha}_1\|_2^2 + L_1\|\tilde{\alpha}_2\|_2^2 + \|\alpha_{1:2}\|_2^2}\right) \quad (2.10)\end{aligned}$$

subject to

$$\sum_{i=1}^{L_1}\alpha_1^i = 0, \quad \sum_{j=1}^{L_2}\alpha_2^j = 0, \quad \sum_{i=1}^{L_1}\tilde{\alpha}_1^i = 0, \quad \sum_{j=1}^{L_2}\tilde{\alpha}_2^j = 0 \quad (2.11)$$

and

$$\sum_{i=1}^{L_1}\alpha_{1:2}^{ij} = 0 \text{ for fixed } j, \quad \sum_{j=1}^{L_2}\alpha_{1:2}^{ij} = 0 \text{ for fixed } i. \quad (2.12)$$

Notice that $\mathbf{X}_i$, $i = 1, 2$ each have two different coefficient vectors α_i and $\tilde{\alpha}_i$, resulting in an overlapped penalty. It follows that the actual main effects θ_1 and θ_2 are given by

$$\theta_1 = \alpha_1 + \tilde{\alpha}_1 \quad (2.13)$$

$$\theta_2 = \alpha_2 + \tilde{\alpha}_2 \quad (2.14)$$

The $\sqrt{L_2\|\tilde{\alpha}_1\|_2^2 + L_1\|\tilde{\alpha}_2\|_2^2 + \|\alpha_{1:2}\|_2^2}$ term results in estimates that satisfy strong hierarchy, because either $\hat{\tilde{\alpha}}_1 = \hat{\tilde{\alpha}}_2 = \hat{\alpha}_{1:2} = 0$ or all are nonzero, i.e. interactions are always present with both main effects.

The constants L_1 and L_2 are chosen to put $\tilde{\alpha}_1$, $\tilde{\alpha}_2$, and $\alpha_{1:2}$ on the same scale. To motivate this, note that we can write

$$X_1\tilde{\alpha}_1 = X_{1:2}[\underbrace{\tilde{\alpha}_1, \ldots, \tilde{\alpha}_1}_{L_2 \text{ copies}}]^T, \quad (2.15)$$

and similarly for $X_2\tilde{\alpha}_2$. We now have a representation for $\tilde{\alpha}_1$ and $\tilde{\alpha}_2$ with respect to the space defined by $X_{1:2}$, so that they are "comparable" to $\alpha_{1:2}$. We then have

$$\|[\underbrace{\tilde{\alpha}_1, \dots, \tilde{\alpha}_1}_{L_2 \text{ copies}}]\|_2^2 = L_2\|\tilde{\alpha}_1\|_2^2 \tag{2.16}$$

and likewise for $\tilde{\alpha}_2$. More details are given in Section 2.3.2 below.

The estimated main effects and interactions can be recovered as

$$\hat{\theta}_1 = \hat{\alpha}_1 + \hat{\tilde{\alpha}}_1 \tag{2.17}$$

$$\hat{\theta}_2 = \hat{\alpha}_2 + \hat{\tilde{\alpha}}_2 \tag{2.18}$$

$$\hat{\theta}_{1:2} = \hat{\alpha}_{1:2}. \tag{2.19}$$

Because of the "all zero" or "all nonzero" property of the group-lasso estimates mentioned above, we also have

$$\hat{\theta}_{1:2} \neq 0 \Longrightarrow \hat{\theta}_1 \neq 0 \text{ and } \hat{\theta}_2 \neq 0. \tag{2.20}$$

The overlapped group-lasso with constraints is conceptually simple, but care must be taken in how we parametrize the constraints. This is especially so because we penalize the coefficients, and any representation of the problem that does not preserve symmetry will result in unequal penalization schemes for the coefficients. The problem becomes more tedious as the number of variables and levels grows. We now show how to solve the overlapped group-lasso problem by solving an equivalent *unconstrained* group-lasso problem. This is advantageous because

1. the problem can be represented in a symmetric way, thus avoiding the need for careful choices of parametrization, and

2. we only have to fit a group-lasso without constraints on the coefficients, which is a well-studied problem.

2.3.2 Equivalence with unconstrained group-lasso

We show that the overlapped group-lasso above can be solved with a simple group-lasso. We will need two Lemmas. The first shows that because we fit an intercept in the model, the estimated coefficients $\hat{\beta}$ for categorical variables will have mean zero.

Lemma 1. *Let X be an indicator matrix. Then the solution $\hat{\beta}$ to*

$$\text{argmin}_{\mu,\beta} \frac{1}{2} \|\mathbf{Y} - \mu \cdot \mathbf{1} - \mathbf{X}\beta\|_2^2 + \lambda \|\beta\|_2 \tag{2.21}$$

satisfies

$$\bar{\hat{\beta}} = 0. \tag{2.22}$$

The same is true for logistic loss.

Proof. Because $\mathbf{X}$ is an indicator matrix, each row consists of exactly a single 1 (all other entries 0), so that

$$\mathbf{X} \cdot c\mathbf{1} = c\mathbf{1} \tag{2.23}$$

for any constant c. It follows that if $\hat{\mu}$ and $\hat{\beta}$ are solutions, then so are $\hat{\mu} + c\mathbf{1}$ and $\hat{\beta} - c\mathbf{1}$. But the norm $\|\hat{\beta} - c\mathbf{1}\|_2$ is minimized for $c = \bar{\hat{\beta}}$. □

The next Lemma states that if we include two intercepts in the model, one penalized and the other unpenalized, then the penalized intercept will be estimated to be zero. This is because we can achieve the same fit with a lower penalty by taking $\mu \longleftarrow \mu + \tilde{\mu}$.

Lemma 2. *The optimization problem*

$$\text{argmin}_{\mu,\tilde{\mu},\beta} \frac{1}{2} \|\mathbf{Y} - \mu \cdot \mathbf{1} - \tilde{\mu} \cdot \mathbf{1} - \ldots\|_2^2 + \lambda \sqrt{\|\tilde{\mu}\|_2^2 + \|\beta\|_2^2} \tag{2.24}$$

has solution $\hat{\tilde{\mu}} = 0$ for all $\lambda > 0$. The same result holds for logistic loss.

The next theorem shows how the overlapped group-lasso in Section 2.3.1 reduces to a group-lasso.

Theorem 1. *Solving the constrained optimization problem (2.10) - (2.12) in Section 2.3.1 is equivalent to solving the unconstrained problem*

$$\operatorname{argmin}_{\mu,\beta} \frac{1}{2} \left\| \mathbf{Y} - \mu \cdot \mathbf{1} - \mathbf{X}_1\beta_1 - \mathbf{X}_2\beta_2 - \mathbf{X}_{1:2}\beta_{1:2} \right\|_2^2 \\ + \lambda \left(\|\beta_1\|_2 + \|\beta_2\|_2 + \|\beta_{1:2}\|_2 \right). \quad (2.25)$$

Proof. We need to show that the group-lasso objective can be equivalently written as an overlapped group-lasso with the appropriate constraints on the parameters. We begin by rewriting (2.10) as

$$\operatorname{argmin}_{\mu,\tilde{\mu},\alpha,\tilde{\alpha}} \frac{1}{2} \left\| \mathbf{Y} - \mu \cdot \mathbf{1} - \mathbf{X}_1\alpha_1 - \mathbf{X}_2\alpha_2 - [\mathbf{1}\ \mathbf{X}_1\ \mathbf{X}_2\ \mathbf{X}_{1:2}] \begin{bmatrix} \tilde{\mu} \\ \tilde{\alpha}_1 \\ \tilde{\alpha}_2 \\ \alpha_{1:2} \end{bmatrix} \right\|_2^2 \\ + \lambda \left(\|\alpha_1\|_2 + \|\alpha_2\|_2 + \sqrt{L_1L_2\|\tilde{\mu}\|_2^2 + L_2\|\tilde{\alpha}_1\|_2^2 + L_1\|\tilde{\alpha}_2\|_2^2 + \|\alpha_{1:2}\|_2^2} \right). \quad (2.26)$$

By Lemma 2, we will estimate $\hat{\tilde{\mu}} = 0$. Therefore we have not changed the solutions in any way.

Lemma 1 shows that the first two constraints in (2.11) are satisfied by the estimated main effects $\hat{\beta}_1$ and $\hat{\beta}_2$. We now show that

$$\|\beta_{1:2}\|_2 = \sqrt{L_1L_2\|\tilde{\mu}\|_2^2 + L_2\|\tilde{\alpha}_1\|_2^2 + L_1\|\tilde{\alpha}_2\|_2^2 + \|\alpha_{1:2}\|_2^2} \quad (2.27)$$

where the $\tilde{\alpha}_1, \tilde{\alpha}_2$, and $\alpha_{1:2}$ satisfy the constraints in (2.11) and (2.12).

For fixed levels i and j, we can decompose $\beta_{1:2}$ (see [35]) as

$$\begin{aligned} \beta_{1:2}^{ij} &= \beta_{1:2}^{\cdot\cdot} + (\beta_{1:2}^{i\cdot} - \beta_{1:2}^{\cdot\cdot}) + (\beta_{1:2}^{\cdot j} - \beta_{1:2}^{\cdot\cdot}) + (\beta_{1:2}^{ij} - \beta_{1:2}^{i\cdot} - \beta_{1:2}^{\cdot j} + \beta_{1:2}^{\cdot\cdot}) \quad (2.28) \\ &\equiv \tilde{\mu} + \tilde{\alpha}_1^i + \tilde{\alpha}_2^j + \alpha_{1:2}^{ij}. \quad (2.29) \end{aligned}$$

It follows that the whole (L_1L_2)-vector $\beta_{1:2}$ can be written as

$$\beta_{1:2} = \mathbf{1}\tilde{\mu} + \mathbf{Z}_1\tilde{\alpha}_1 + \mathbf{Z}_2\tilde{\alpha}_2 + \alpha_{1:2}, \tag{2.30}$$

where $\mathbf{Z}_1$ is a $L_1L_2 \times L_1$ indicator matrix of the form

$$\underbrace{\begin{pmatrix} \mathbf{1}_{L_2\times 1} & 0 & \cdots & 0 \\ 0 & \mathbf{1}_{L_2\times 1} & \cdots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \mathbf{1}_{L_2\times 1} \end{pmatrix}}_{L_1 \text{ columns}} \tag{2.31}$$

and $\mathbf{Z}_2$ is a $L_1L_2 \times L_2$ indicator matrix of the form

$$L_1 \text{ copies} \left\{ \begin{pmatrix} I_{L_2\times L_2} \\ \vdots \\ I_{L_2\times L_2} \end{pmatrix} \right. \tag{2.32}$$

It follows that

$$\mathbf{Z}_1\tilde{\alpha}_1 = (\underbrace{\tilde{\alpha}_1^1, \ldots, \tilde{\alpha}_1^1}_{L_2 \text{ copies}}, \underbrace{\tilde{\alpha}_1^2, \ldots, \tilde{\alpha}_1^2}_{L_2 \text{ copies}}, \ldots, \underbrace{\tilde{\alpha}_1^{L_1}, \ldots, \tilde{\alpha}_1^{L_1}}_{L_2 \text{ copies}})^T \tag{2.33}$$

and

$$\mathbf{Z}_2\tilde{\alpha}_2 = (\tilde{\alpha}_2^1, \ldots, \tilde{\alpha}_2^{L_2}, \tilde{\alpha}_2^1, \ldots, \tilde{\alpha}_2^{L_2}, \ldots, \tilde{\alpha}_2^1, \ldots, \tilde{\alpha}_2^{L_2})^T. \tag{2.34}$$

Note that $\tilde{\alpha}_1, \tilde{\alpha}_2$, and $\alpha_{1:2}$, by definition, satisfy the constraints (2.11) and (2.12). This can be used to show, by direct calculation, that the four additive components in (2.30) are mutually orthogonal, so that we can write

$$\begin{aligned} \|\beta_{1:2}\|_2^2 &= \|\mathbf{1}\tilde{\mu}\|_2^2 + \|\mathbf{Z}_1\tilde{\alpha}_1\|_2^2 + \|\mathbf{Z}_2\tilde{\alpha}_2\|_2^2 + \|\alpha_{1:2}\|_2^2 && (2.35) \\ &= L_1L_2\|\tilde{\mu}\|_2^2 + L_2\|\tilde{\alpha}_1\|_2^2 + L_1\|\tilde{\alpha}_2\|_2^2 + \|\alpha_{1:2}\|_2^2. && (2.36) \end{aligned}$$

We have shown that the penalty in the group-lasso problem is equivalent to the penalty in the constrained overlapped group-lasso. It remains to show that the loss functions in both problems are also the same. Since $\mathbf{X}_{1:2}\mathbf{Z}_1 = \mathbf{X}_1$ and $\mathbf{X}_{1:2}\mathbf{Z}_2 = \mathbf{X}_2$, this can be seen by a direct computation:

$$\begin{aligned}
\mathbf{X}_{1:2}\beta_{1:2} &= \mathbf{X}_{1:2}(\mathbf{1}\tilde{\mu} + \mathbf{Z}_1\tilde{\alpha}_1 + \mathbf{Z}_2\tilde{\alpha}_2 + \alpha_{1:2}) && (2.37)\\
&= \mathbf{1}\tilde{\mu} + \mathbf{X}_1\tilde{\alpha}_1 + \mathbf{X}_2\tilde{\alpha}_2 + \mathbf{X}_{1:2}\alpha_{1:2} && (2.38)\\
&= [\mathbf{1}\ \mathbf{X}_1\ \mathbf{X}_2\ \mathbf{X}_{1:2}] \begin{bmatrix} \tilde{\mu} \\ \tilde{\alpha}_1 \\ \tilde{\alpha}_2 \\ \alpha_{1:2} \end{bmatrix} && (2.39)
\end{aligned}$$

□

Theorem 1 shows that we can use the group-lasso to obtain estimates that satisfy strong hierarchy, without solving the overlapped group-lasso with constraints. The theorem also shows that the main effects and interactions can be extracted with

$$\begin{aligned}
\hat{\theta}_1 &= \hat{\beta}_1 + \hat{\tilde{\alpha}}_1 && (2.40)\\
\hat{\theta}_2 &= \hat{\beta}_2 + \hat{\tilde{\alpha}}_2 && (2.41)\\
\hat{\theta}_{1:2} &= \hat{\alpha}_{1:2}. && (2.42)
\end{aligned}$$

We discuss the properties of the GLINTERNET estimates in the next section.

2.3.3 Properties of the glinternet estimators

While GLINTERNET treats the problem as a group-lasso, examining the equivalent overlapped group-lasso version makes it easier to draw insights about the behaviour of the method under various scenarios. Recall that the overlapped penalty for two variables is given by

$$\|\alpha_1\|_2 + \|\alpha_2\|_2 + \sqrt{L_2\|\tilde{\alpha}_1\|_2^2 + L_1\|\tilde{\alpha}_2\|_2^2 + \|\alpha_{1:2}\|_2^2}. \qquad (2.43)$$

If the ground truth is additive, i.e. $\alpha_{1:2} = 0$, then $\tilde{\alpha}_1$ and $\tilde{\alpha}_2$ will be estimated to be zero (in the noiseless case). This is because for $L_1, L_2 \geq 2$ and $a, b \geq 0$, we have

$$\sqrt{L_2 a^2 + L_1 b^2} \geq a + b. \tag{2.44}$$

Thus it is advantageous to place all the main effects in α_1 and α_2, because doing so results in a smaller penalty. Therefore, if the truth has no interactions, then GLINTERNET picks out only main effects.

If an interaction was present ($\alpha_{1:2} > 0$), the derivative of the penalty term with respect to $\alpha_{1:2}$ is

$$\frac{\alpha_{1:2}}{\sqrt{L_2 \|\tilde{\alpha}_1\|_2^2 + L_1 \|\tilde{\alpha}_2\|_2^2 + \|\alpha_{1:2}\|_2^2}}. \tag{2.45}$$

The presence of main effects allows this derivative to be smaller, thus allowing the algorithm to pay a smaller penalty (as compared to no main effects present) for making $\hat{\alpha}_{1:2}$ nonzero. This shows interactions whose main effects are also present are discovered before pure interactions.

2.3.4 Interaction between a categorical variable and a continuous variable

We describe how to extend Theorem 1 to interaction between a continuous variable and a categorical variable.

Consider the case where we have a categorical variable F with L levels, and a continuous variable Z. Let $\mu_i = \mathbb{E}[Y|F = i, Z = z]$. There are four cases:

- $\mu_i = \mu$ (no main effects, no interactions)
- $\mu_i = \mu + \theta_1^i$ (main effect F)
- $\mu_i = \mu + \theta_1^i + \theta_2 z$ (two main effects)
- $\mu_i = \mu + \theta_1^i + \theta_2 z + \theta_{1:2}^i z$ (main effects and interaction)

As before, we impose the constraints $\sum_{i=1}^{L} \theta_1^i = 0$ and $\sum_{i=1}^{L} \theta_{1:2}^i = 0$. An overlapped group-lasso of the form

$$\operatorname{argmin}_{\mu,\alpha,\tilde{\alpha}} \frac{1}{2} \left\| \mathbf{Y} - \mu \cdot \mathbf{1} - \mathbf{X}\alpha_1 - \mathbf{Z}\alpha_2 - [\mathbf{X} \quad \mathbf{Z} \quad (\mathbf{X} * \mathbf{Z})] \begin{bmatrix} \tilde{\alpha}_1 \\ \tilde{\alpha}_2 \\ \alpha_{1:2} \end{bmatrix} \right\|_2^2 \\ + \lambda \left(\|\alpha_1\|_2 + \|\alpha_2\|_2 + \sqrt{\|\tilde{\alpha}_1\|_2^2 + L\|\tilde{\alpha}_2\|_2^2 + \|\alpha_{1:2}\|_2^2} \right) \quad (2.46)$$

subject to

$$\sum_{i=1}^{L} \alpha_1^i = 0, \quad \sum_{i=1}^{L} \tilde{\alpha}_1^i = 0, \quad \sum_{i=1}^{L} \alpha_{1:2}^i = 0 \quad (2.47)$$

allows us to obtain estimates of the interaction term that satisfy strong hierarchy. This is again due to the nature of the square root term in the penalty. The actual main effects and interactions can be recovered as

$$\hat{\theta}_1 = \hat{\alpha}_1 + \hat{\tilde{\alpha}}_1 \quad (2.48)$$

$$\hat{\theta}_2 = \hat{\alpha}_2 + \hat{\tilde{\alpha}}_2 \quad (2.49)$$

$$\hat{\theta}_{1:2} = \hat{\alpha}_{1:2}. \quad (2.50)$$

We have the following extension of Theorem 1:

Theorem 2. *Solving the constrained overlapped group-lasso above is equivalent to solving*

$$\operatorname{argmin}_{\mu,\beta} \frac{1}{2} \|\mathbf{Y} - \mu \cdot \mathbf{1} - \mathbf{X}\beta_1 - \mathbf{Z}\beta_2 - (\mathbf{X} * [\mathbf{1} \quad \mathbf{Z}])\beta_{1:2}\|_2^2 \\ + \lambda \left(\|\beta_1\|_2 + \|\beta_2\|_2 + \|\beta_{1:2}\|_2 \right). \quad (2.51)$$

Proof. We proceed as in the proof of Theorem 1 and introduce an additional parameter $\tilde{\mu}$ into the overlapped objective:

$$\text{argmin}_{\mu,\tilde{\mu},\alpha,\tilde{\alpha}} \frac{1}{2} \left\| \mathbf{Y} - \mu \cdot \mathbf{1} - \mathbf{X}\alpha_1 - \mathbf{Z}\alpha_2 - [\mathbf{1} \quad \mathbf{X} \quad \mathbf{Z} \quad (\mathbf{X} * \mathbf{Z})] \begin{bmatrix} \tilde{\mu} \\ \tilde{\alpha}_1 \\ \tilde{\alpha}_2 \\ \alpha_{1:2} \end{bmatrix} \right\|_2^2$$

$$+ \lambda \left(\|\alpha_1\|_2 + \|\alpha_2\|_2 + \sqrt{L\|\tilde{\mu}\|_2^2 + \|\tilde{\alpha}_1\|_2^2 + L\|\tilde{\alpha}_2\|_2^2 + \|\alpha_{1:2}\|_2^2} \right) \quad (2.52)$$

As before, this does not change the solutions because we will have $\hat{\tilde{\mu}} = 0$ (see Lemma 2).

Decompose the $2L$-vector $\beta_{1:2}$ into

$$\begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix}, \quad (2.53)$$

where η_1 and η_2 both have dimension $L \times 1$. Apply the anova decomposition to both to obtain

$$\begin{aligned} \eta_1^i &= \eta_1^{\cdot} + (\eta_1^i - \eta_1^{\cdot}) && (2.54) \\ &\equiv \tilde{\mu} + \tilde{\alpha}_1^i && (2.55) \end{aligned}$$

and

$$\begin{aligned} \eta_2^i &= \eta_2^{\cdot} + (\eta_2^i - \eta_2^{\cdot}) && (2.56) \\ &\equiv \tilde{\alpha}_2 + \alpha_{1:2}^i. && (2.57) \end{aligned}$$

Note that $\tilde{\alpha}_1$ is a $(L \times 1)$-vector that satisfies $\sum_{i=1}^{L} \tilde{\alpha}_2^i = 0$, and likewise for $\alpha_{1:2}$. This allows us to write

$$\beta_{1:2} = \begin{bmatrix} \tilde{\mu} \cdot \mathbf{1}_{L\times 1} \\ \tilde{\alpha}_2 \cdot \mathbf{1}_{L\times 1} \end{bmatrix} + \begin{bmatrix} \tilde{\alpha}_1 \\ \alpha_{1:2} \end{bmatrix} \quad (2.58)$$

It follows that

$$\|\beta_{1:2}\|_2^2 = L\|\tilde{\mu}\|_2^2 + \|\tilde{\alpha}_1\|_2^2 + L\|\tilde{\alpha}_2\|_2^2 + \|\alpha_{1:2}\|_2^2, \tag{2.59}$$

which shows that the penalties in both problems are equivalent. A direct computation shows that the loss functions are also equivalent:

$$\begin{aligned}
(\mathbf{X} * [\mathbf{1} \quad \mathbf{Z}])\beta_{1:2} &= [\mathbf{X} \quad (\mathbf{X} * \mathbf{Z})]\beta_{1:2} && (2.60)\\
&= [\mathbf{X} \quad (\mathbf{X} * \mathbf{Z})] \left(\begin{bmatrix} \tilde{\mu} \cdot \mathbf{1}_{L\times 1} \\ \tilde{\alpha}_2 \cdot \mathbf{1}_{L\times 1} \end{bmatrix} + \begin{bmatrix} \tilde{\alpha}_1 \\ \alpha_{1:2} \end{bmatrix} \right) && (2.61)\\
&= \tilde{\mu} \cdot \mathbf{1} + \mathbf{X}\tilde{\alpha}_1 + \mathbf{Z}\tilde{\alpha}_2 + (\mathbf{X} * \mathbf{Z})\alpha_{1:2}. && (2.62)
\end{aligned}$$

□

Theorem 2 allows us to accommodate interactions between continuous and categorical variables by simply parametrizing the interaction term as $\mathbf{X} * [\mathbf{1} \quad \mathbf{Z}]$, where $\mathbf{X}$ is the indicator matrix representation for categorical variables that we have been using all along. We then proceed as before with a group-lasso.

2.3.5 Interaction between two continuous variables

We have seen that the appropriate representations for the interaction terms are

- $\mathbf{X}_1 * \mathbf{X}_2 = \mathbf{X}_{1:2}$ for categorical variables
- $\mathbf{X} * [\mathbf{1} \quad \mathbf{Z}] = [\mathbf{X} \quad (\mathbf{X} * \mathbf{Z})]$ for one categorical variable and one continuous variable.

How should we represent the interaction between two continuous variables? Let Z_1 and Z_2 be two continuous variables. One might guess by now that the appropriate form of the interaction term is given by

$$\begin{aligned}
\mathbf{Z}_{1:2} &= [\mathbf{1} \quad \mathbf{Z}_1] * [\mathbf{1} \quad \mathbf{Z}_2] && (2.63)\\
&= [\mathbf{1} \quad \mathbf{Z}_1 \quad \mathbf{Z}_2 \quad (\mathbf{Z}_1 * \mathbf{Z}_2)]. && (2.64)
\end{aligned}$$

This is indeed the case. A linear interaction model for $\mathbf{Z}_1$ and $\mathbf{Z}_2$ is given by

$$\mathbb{E}[Y|Z_1 = z_1, Z_2 = z_2] = \mu + \theta_1 z_1 + \theta_2 z_2 + \theta_{1:2} z_1 z_2. \tag{2.65}$$

Unlike the previous cases where there were categorical variables, there are no constraints on any of the coefficients. It follows that the overlapped group-lasso

$$\begin{aligned} \operatorname{argmin}_{\mu,\tilde{\mu},\alpha,\tilde{\alpha}} \frac{1}{2} \left\| \mathbf{Y} - \mu \cdot \mathbf{1} - \mathbf{Z}_1\alpha_1 - \mathbf{Z}_2\alpha_2 - [\mathbf{1} \quad \mathbf{Z}_1 \quad \mathbf{Z}_2 \quad (\mathbf{Z}_1 * \mathbf{Z}_2)] \begin{bmatrix} \tilde{\mu} \\ \tilde{\alpha}_1 \\ \tilde{\alpha}_2 \\ \alpha_{1:2} \end{bmatrix} \right\|_2^2 \\ + \lambda \left(\|\alpha_1\|_2 + \|\alpha_2\|_2 + \sqrt{\|\tilde{\mu}\|_2^2 + \|\tilde{\alpha}_1\|_2^2 + \|\tilde{\alpha}_2\|_2^2 + \|\alpha_{1:2}\|_2^2} \right) \end{aligned} \tag{2.66}$$

is trivially equivalent to

$$\begin{aligned} \operatorname{argmin}_{\mu,\beta} \frac{1}{2} \|\mathbf{Y} - \mu \cdot \mathbf{1} - \mathbf{Z}_1\beta_1 - \mathbf{Z}_2\beta_2 - ([\mathbf{1} \quad \mathbf{Z}_1] * [\mathbf{1} \quad \mathbf{Z}_2])\beta_{1:2}\|_2^2 \\ + \lambda \left(\|\beta_1\|_2 + \|\beta_2\|_2 + \|\beta_{1:2}\|_2 \right), \end{aligned} \tag{2.67}$$

with the β's taking the place of the α's. Note that we will have $\hat{\tilde{\mu}} = 0$.

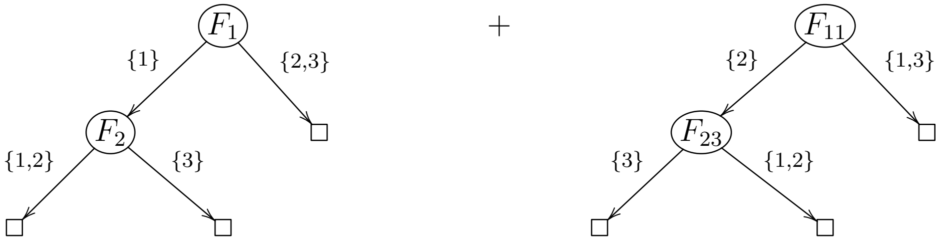
2.4 Variable screening

GLINTERNET works by solving a group-lasso with $p + \binom{p}{2}$ groups of variables. Even for moderate p ($\sim 10^5$), we will require some form of screening to reduce the dimension of the interaction search space. We have argued that models satisfying hierarchy make sense, so that it is natural to consider screening devices that hedge on the presence of main effects. We discuss two screening methods in this section: gradient boosting, and an adaptive screen based on the strong rules of [41]. We describe the boosting approach first.

2.4.1 Screening with boosted trees

AdaBoost [11] and gradient boosting [12] are effective approaches to building ensembles of weak learners such as decision trees. One of the advantages of trees is that they are able to model nonlinear effects and high-order interactions. For example, a depth-2 tree essentially represents an interaction between the variables involved in the two splits, which suggests that boosting with depth-2 trees is a way of building a first-order interaction model. Note that the interactions are hierarchical, because in finding the optimal first split, the boosting algorithm is looking for the best main effect. The subsequent split is then made, conditioned on the first split.

If we boost with T trees, then we end up with a model that has at most T interaction pairs. The following diagram gives a schematic of the boosting iterations with categorical variables.



(2.68)

In the first tree, levels 2 and 3 of F_1 are not involved in the interaction with F_2. Therefore each tree in the boosted model does not represent an interaction among all the levels of the two variables, but only among a subset of the levels. To enforce the full interaction structure, one could use fully-split trees, but we do not develop this approach for two reasons. First, boosting is a sequential procedure and is quite slow even for moderately sized problems. Using fully split trees will further degrade its runtime. Second, in variables with many levels, it is reasonable to expect that the interactions only occur among a few of the levels. If this were true, then a complete interaction that is weak for every combination of levels might be selected over a strong partial interaction. But it is the strong partial interaction that we are interested in.

Boosting is feasible because it is a greedy algorithm. If p is the number of variables, an exhaustive search involves $\mathcal{O}(p^2)$ variables, whereas boosting operates with $\mathcal{O}(p)$. To use the boosted model as a screening device for interaction candidates, we take the set of all unique interactions from the collection of trees. For example, in our schematic above, we would add $F_{1:2}$ and $F_{11:23}$ to our candidate set of interactions.

In our experiments, using boosting as a screen did not perform as well as we hoped. There is the issue of selecting tuning parameters such as the amount of shrinkage and the number of trees to use. Lowering the shrinkage and increasing the number of trees improves false discovery rates, but at a significant cost to speed. In the next section, we describe a screening approach that is based on computing inner products that is efficient and that can be integrated with the strong rules for the group-lasso.

2.4.2 An adaptive screening procedure

The strong rules [41] for lasso-type problems are effective heuristics for discarding large numbers of variables that are likely to be redundant. As a result, the strong rules can dramatically speed up the convergence of algorithms because they can concentrate on a smaller set (we call this the *strong set*) of variables that are more likely to be nonzero. The strong rules are not safe, however, meaning that it is possible that some of the discarded variables are actually supposed to be nonzero. Because of this, after our algorithm has converged on the strong set, we have to check the KKT conditions on the discarded set. Those variables that do not satisfy the conditions then have to be added to the current set of nonzero variables, and we fit on this expanded set. This happens rarely in our experience, i.e. the discarded variables tend to remain zero after the algorithm has converged on the strong set, which means we rarely have to do multiple rounds of fitting for any given value of the regularization paramter λ.

The strong rules for the group-lasso involve computing $s_i = \|\mathbf{X}_i^T(\mathbf{Y} - \hat{\mathbf{Y}})\|_2$ for every group of variables $\mathbf{X}_i$, and then discarding a group i if $s_i < 2\lambda_{current} - \lambda_{previous}$. If this is feasible for all $p + \binom{p}{2}$ groups, then there is no need for screening; we simply fit the group-lasso on those groups that pass the strong rules filter. Otherwise, we approximate this by screening only on the groups that correspond to main effects. We then take the candidate set of interactions to consist of all pairwise interactions between the variables that passed this screen. Note that because the KKT conditions for group i are (see Section 1.2)

$$s_i < \lambda \quad \text{if} \quad \hat{\beta}_i = 0 \tag{2.69}$$

$$s_i = \lambda \quad \text{if} \quad \hat{\beta}_i \neq 0, \tag{2.70}$$

we will have already computed the s_i for the strong rules from checking the KKT conditions for the solutions at the previous λ. This allows us to integrate screening with the strong rules in an efficient manner. An example will illustrate.

Suppose we have 10,000 variables ($\sim 50 \times 10^6$ possible interactions), but we are computationally limited to a group-lasso with 10^6 groups. Assume we have the fit for $\lambda = \lambda_k$, and want to move on to λ_{k+1}. Let $r_{\lambda_k} = \mathbf{Y} - \hat{\mathbf{Y}}_{\lambda_k}$ denote the current residual. At this point, the variable scores $s_i = \|\mathbf{X}_i^T r_{\lambda_k}\|_2$ have already been computed from checking the KKT conditions at the solutions for λ_k. We restrict ourselves to the 10,000 variables, and take the 100 with the highest scores. Denote this set by $\mathcal{T}_{100}^{\lambda_{k+1}}$. The candidate set of variables for the group-lasso is then given by $\mathcal{T}_{100}^{\lambda_{k+1}}$ together with the pairwise interactions between *all* 10,000 variables and $\mathcal{T}_{100}^{\lambda_{k+1}}$. Because this gives a candidate set with about $100 \times 10,000 = 10^6$ terms, the compuation is now feasible. We then compute the group-lasso on this candidate set, and repeat the procudure with the new residual r_{λ_k+1}.

This screen is easy to compute since it is based on inner products. Moreover, they can be computed in parallel. The procedure also integrates well with the strong rules by reusing inner products computed from the fit for a previous λ.

2.5 Related work and approaches

We describe some past and related approaches to discovering interactions. We give a short synopsis of how they work, and say why they are inadequate for our purposes. The method most similar to ours is hierNet.

2.5.1 Logic regression [33]

Logic regression finds boolean combinations of variables that have high predictive power of the response variable. For example, a combination might look like

$$(F_1 \text{ and } F_3) \text{ or } F_5. \tag{2.71}$$

This is an example of an interaction that is of higher-order than what GLINTERNET handles, and is an appealing aspect of logic regression. However, logic regression does not accommodate continuous variables or categorical variables with more than two levels. We do not make comparisons with logic regression in our simulations for this reason.

2.5.2 Composite absolute penalties [50]

Like GLINTERNET, this is also a penalty-based approach. CAP employs penalties of the form

$$\|(\beta_i, \beta_j)\|_{\gamma_1} + \|\beta_j\|_{\gamma_2} \tag{2.72}$$

where $\gamma_1 > 1$. Such a penalty ensures that $\hat{\beta}_i \neq 0$ whenever $\hat{\beta}_j \neq 0$. It is possible that $\hat{\beta}_i \neq 0$ but $\hat{\beta}_j = 0$. In other words, the penalty makes $\hat{\beta}_j$ hierarchically dependent on $\hat{\beta}_i$: it can only be nonzero after $\hat{\beta}_i$ becomes nonzero. It is thus possible to use CAP penalties to build interaction models that satisfy hierarchy. For example, a penalty of the form $\|(\theta_1, \theta_2, \theta_{1:2})\|_2 + \|\theta_{1:2}\|_2$ will result in estimates that satisfy $\hat{\theta}_{1:2} \neq 0 \Longrightarrow \hat{\theta}_1 \neq 0$ and $\hat{\theta}_2 \neq 0$. We can thus build a linear interaction model for two categorical variables by solving

$$\begin{aligned}\text{argmin}_{\mu,\theta} \frac{1}{2} \|\mathbf{Y} - \mu \cdot \mathbf{1} - \mathbf{X}_1\theta_1 - \mathbf{X}_2\theta_2 - \mathbf{X}_{1:2}\theta_{1:2}\|_2^2 \\ + \lambda(\|(\theta_1, \theta_2, \theta_{1:2})\|_2 + \|\theta_{1:2}\|_2)\end{aligned} \tag{2.73}$$

subject to (2.3) and (2.4). We see that the CAP approach differs from GLINTERNET in that we have to solve a constrained optimization problem. The form of the penalties are also different: the interaction coefficient in CAP is penalized twice, whereas GLINTERNET penalizes it once. This, together with the constraints, results in a more difficult optimization problem, and it is not obvious what the relationship between the two algorithms' solutions would be.

2.5.3 hierNet [5]

This is a method that, like GLINTERNET, seeks to find interaction estimates that obey hierarchy with regularization. The optimization problem that hierNet solves is

$$\operatorname{argmin}_{\mu,\beta,\theta} \frac{1}{2}\sum_{i=1}^{n}(y_i - \mu - x_i^T\beta - \frac{1}{2}x_i^T\theta x_i)^2 + \lambda\mathbf{1}^T(\beta^+ + \beta^-) + \frac{\lambda}{2}\|\theta\|_1 \qquad (2.74)$$

subject to

$$\theta = \theta^T, \|\theta_j\|_1 \leq \beta_j^+ + \beta_j^-, \beta_j^+ \geq 0, \beta_j^- \geq 0. \qquad (2.75)$$

The main effects are represented by β, and interactions are given by θ. The first constraint enforces symmetry in the interaction coefficients. β_j^+ and β_j^- are the positive and negative parts of β_j, and are given by $\beta_j^+ = \max(0, \beta_j)$ and $\beta_j^- = -\min(0, \beta_j)$ respectively. The constraint $\|\theta_j\|_1 \leq \beta_j^+ + \beta_j^-$ implies that if some components of the j-th row of θ are estimated to be nonzero, then the main effect β_j will also be estimated to be nonzero. Since θ_j corresponds to interactions between the j-th variable and all the other variables, this implies that the solutions to the hierNet objective satisfy weak hierarchy. One can think of $\beta_j^+ + \beta_j^-$ as a budget for the amount of interactions that are allowed to be nonzero.

The hierNet objective can be modified to obtain solutions that satisfy strong hierarchy, which makes it in principle comparable to GLINTERNET. Currently, hierNet is only able to accommodate binary and continuous variables, and is practically limited to fitting models with fewer than 1000 variables.

2.6 Simulation study

We perform simulations to see if GLINTERNET is competitive with existing methods. hierNet is a natural benchmark because it also tries to find interactions subject to hierarchical constraints. Because hierNet only works with continuous variables and 2-level categorical variables, we include gradient boosting as a competitor for the scenarios where hierNet cannot be used.

2.6.1 False discovery rates

We simulate 4 different setups:

1. Truth obeys strong hierarchy. The interactions are only among pairs of nonzero main effects.

2. Truth obeys weak hierarchy. Each interaction has only one of its main effects present.

3. Truth is anti-hierarchical. The interactions are only among pairs of main effects that are not present.

4. Truth is pure interaction. There are no main effects present, only interactions.

Each case is generated with $n = 500$ observations and $p = 30$ continuous variables, with a signal to noise ratio of 1. Where applicable, there are 10 main effects and/or 10 interactions in the ground truth. The interaction and main effect coefficients are sampled from $N(0, 1)$, so that the variance in the observations should be split equally between main effects and interactions.

Boosting is done with 5000 depth-2 trees and a learning rate of 0.001. Each tree represents a candidate interaction, and we can compute the improvement to fit due to this candidate pair. Summing up the improvement over the 5000 trees gives a score for each interaction pair, which can then be used to order the pairs. We then compute the false discovery rate as a function of rank. For GLINTERNET and hierNet, we obtain a path of solutions and compute the false discovery rate as a function of the number of interactions discovered. The default setting for hierNet is to impose weak hierarchy, and we use this except in the cases where the ground truth has strong hierarchy. In these cases, we set hierNet to impose strong hierarchy. We also set "diagonal=FALSE" to disable quadratic terms.

We plot the average false discovery rate with standard error bars as a function of the number of predicted interactions in Figure 2.2. The results are from 100 simulation runs. We see that GLINTERNET is competitive with hierNet when the truth obeys strong or weak hierarchy, and does better when the truth is anti-hierarchical. This is expected because hierNet requires the presence of main effects as a budget for interactions, whereas GLINTERNET can still esimate an interaction to be nonzero even though none of its main effects are present. Boosting is not competitive, especially in the anti-hierarchical case. This is because the first split in a tree is effectively looking for a main effect.

Both GLINTERNET and hierNet perform comparably in these simulations. If all the variables are continuous, there do not seem to be compelling reasons to choose one over the other.

2.6.2 Feasibility

To the best of our knowledge, hierNet is the only readily available package for learning interactions among continuous variables in a hierarchical manner. Therefore it is natural to use hierNet as a speed benchmark. We generate data in which the ground truth has strong hierarchy as in Section 2.6.1, but with $n = 1000$ quantitative observations and $p = 20, 40, 80, 160, 320, 640$ continuous variables. We set each method to find 10 interactions. While hierNet does not allow the user to specify the number of interactions to discover, we get around this by fitting a path of values, then selecting the regularization parameter that corresponds to 10 nonzero estimated interactions. We then refit hierNet along a path that terminates with this choice of parameter, and time this run. Both software packages are compiled with the same options. Figure 2.3 shows the best time recorded for each method over 10 runs. These simulations

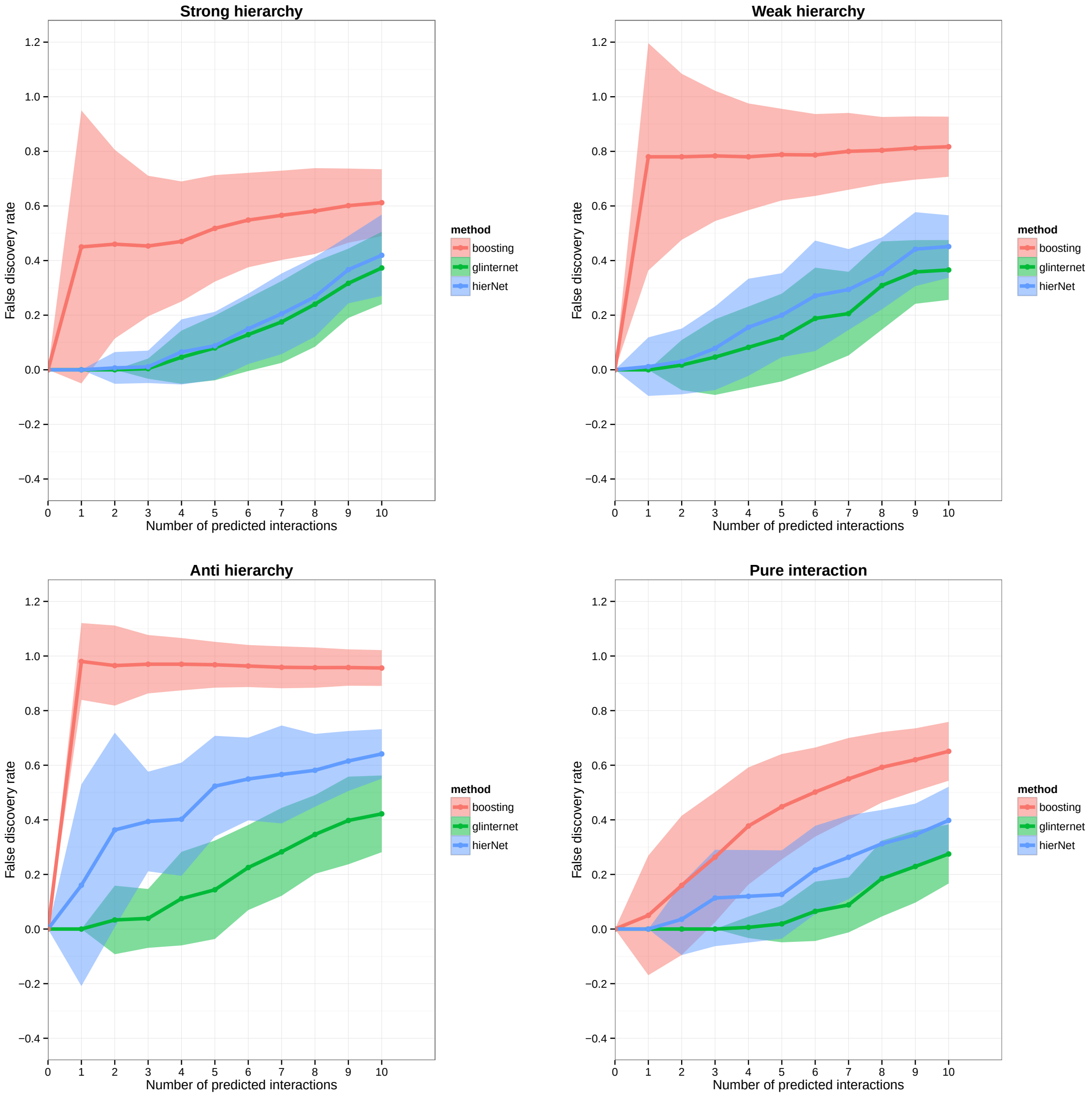


Figure 2.2: Simulation results for continuous variables: Average false discovery rate and standard errors from 100 simulation runs.

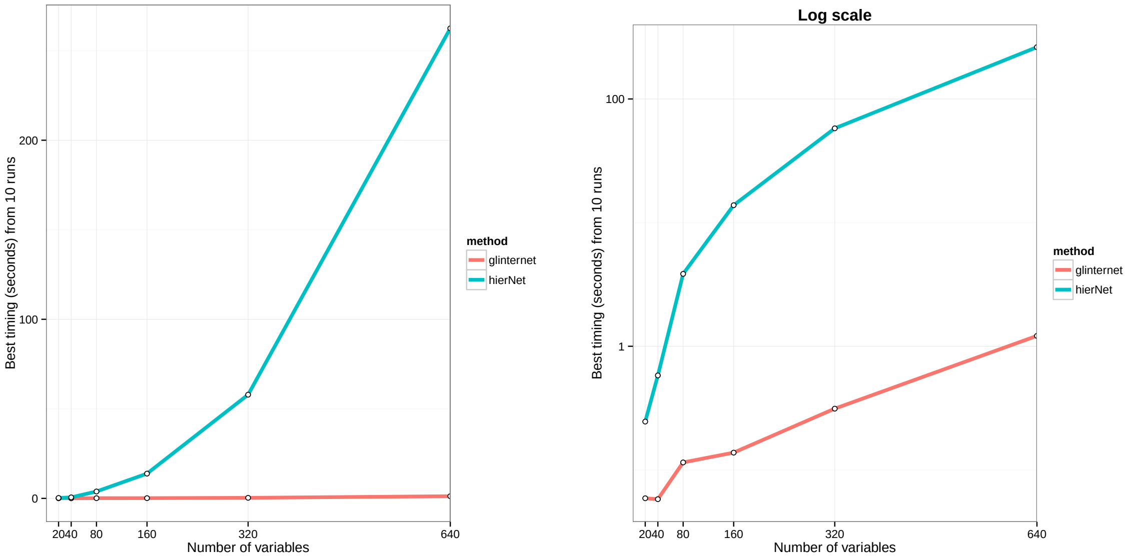


Figure 2.3: **Left:** Best wallclock time over 10 runs for discovering 10 interactions. **Right:** log scale.

were timed on a Intel Core-i7 3930K processor.

2.7 Real data examples

We compare the performance of GLINTERNET on several prediction problems. The competitor methods used are gradient boosting, lasso, ridge regression, and hierNet where feasible. In all situations, we determine the number of trees in boosting by first building a model with a large number of trees, typically 5000 or 10000, and then selecting the number that gives the lowest cross-validated error. We use a learning rate of 0.001, and we do not subsample the data since the sample sizes are small in all the cases.

The methods are evaluated on three measures:

1. missclassification error, or 0-1 loss

2. area under the receiver operating characteristic (ROC) curve, or auc

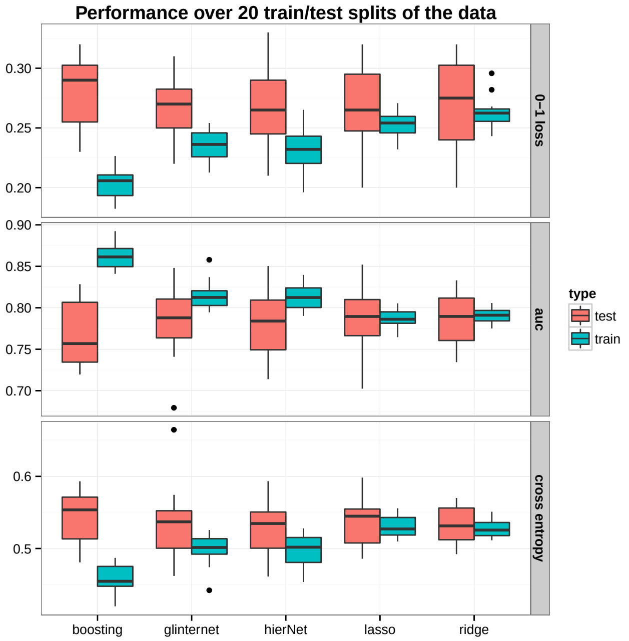


Figure 2.4: Performance of methods on 20 train-test splits of the South African heart disease data.

3. cross entropy, given by $-\frac{1}{n}\sum_{i=1}^{n}[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)]$.

2.7.1 South African heart disease data

The data consists of 462 males from a high risk region for heart disease in South Africa. The task is to predict which subjects had coronary heart disease using risk factors such as cumulative tobacco, blood pressure, and family history of heart disease. We randomly split the data into 362-100 train-test examples, and tuned each method on the training data using 10-fold cross validation before comparing the prediction performance on the held out test data. This splitting process was carried out 20 times, and Figure 2.4 summarizes the results. The methods are all comparable, with no distinct winner.

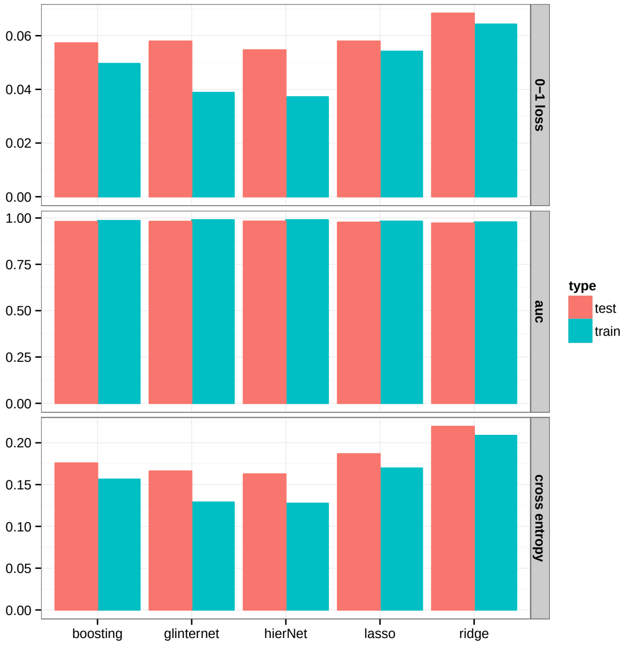


Figure 2.5: Performance on the Spambase data.

2.7.2 Spambase

This is the Spambase data taken from the UCI Machine Learning Repository. There are 4601 binary observations indicating whether an email is spam or non-spam, and 57 integer-valued variables. All the features are log-transformed by $\log(1 + x)$ before applying the methods. We split the data into a training set consisting of 3065 observations and a test set consisting of 1536 observations. The methods are tuned on the training set using 10-fold cross validation before predicting on the test set. The results are shown in Figure 2.5.

2.7.3 Dorothea

Dorothea is one of the 5 datasets from the NIPS 2003 Feature Learning Challenge, where the goal is to predict if a chemical molecule will bind to a receptor target. There are 100000 binary features that describe three-dimensional properties of the molecules, half of which are probes that have nothing to do with the response. The training, validation, and test sets consist of 800, 350, and 800 observations respectively. More details about how the data were prepared can be found at http://archive.ics.uci.edu/ml/datasets/Dorothea.

We run GLINTERNET with screening on 1000 main effects, which results in about 100 million candidate interaction pairs. The validation set was used to tune all the methods. We then predict on the test set with the chosen models and submitted the results online for scoring. The best model chosen by GLINTERNET made use of 93 features, compared with the 9 features chosen by L_1-penalized logistic regression (lasso). Figure 2.6 summarizes the performance for each method. We see that GLINTERNET has a slight advantage over the lasso, indicating that interactions might be important for this problem. Boosting did not perform well in our false discovery rate simulations, which could be one of the reasons why it does not do well here despite taking interactions into account.

2.7.4 Genome-wide association study

We use the simulated rheumatoid arthritis data (replicate 1) from Problem 3 in Genetic Analysis Workshop 15. Affliction status was determined by a genetic/environmental model to mimic the familial pattern of arthritis; full details can be found in [28]. The authors simulated a large population of nuclear families consisting of two parents and two offspring. We are then provided with 1500 randomly chosen families with an affected sibling pair (ASP), and 2,000 unaffected families as a control group. For the control families, we only have data from one randomly chosen sibling. Therefore we also sample one sibling from each of the 1,500 ASPs to obtain 1,500 cases.

There are 9,187 single nucleotide polymorphism (SNP) markers on chromosomes 1 through 22 that are designed to mimic a 10K SNP chip set, and a dense set of 17,820 SNPs on chromosome 6 that approximate the density of a 300K SNP set. Since 210 of the SNPs on chromosome 6 are found in both the dense and non-dense sets, we made sure to include them only once in our analysis. This gives us a total of

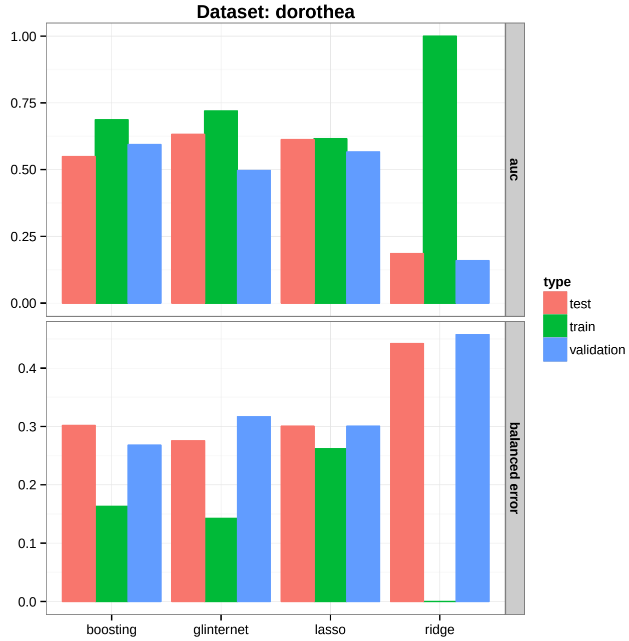


Figure 2.6: Performance on dorothea

$9187 - 210 + 17820 = 26797$ SNPs, all of which are 3-level categorical variables.

We are also provided with phenotype data, and we include sex, age, smoking history, and the DR alleles from father and mother in our analysis. Sex and smoking history are 2-level categorical variables, while age is continuous. Each DR allele is a 3-level categorical variable, and we combine the father and mother alleles in an unordered way to obtain a 6-level DR variable. In total, we have 26,801 variables and 3,500 training examples.

We run GLINTERNET (without screening) on a grid of values for λ that starts with the empty model. The first two variables found are main effects:

- SNP6_305
- denseSNP6_6873

Following that, an interaction denseSNP6_6881:denseSNP6_6882 gets picked up. We now proceed to analyze this result.

Two of the interactions listed in the answer sheet provided with the data are: locus A with DR, and locus C with DR. There is also a main effect from DR. The closest SNP to the given position of locus A is SNP16_31 on chromosome 16, so we take this SNP to represent locus A. The given position for locus C corresponds to denseSNP6_3437, and we use this SNP for locus C. While it looks like none of the true variables are to be found in the list above, these discovered variables have very strong association with the true variables.

If we fit a linear logistic regression model with our first discovered pair denseSNP6_6881:denseSNP6_6882, the main effect denseSNP6_6882 and the interaction terms are both significant:

	Df	Dev	Resid. Dev	$\mathbb{P}(>\chi^2)$
NULL			4780.4	
denseSNP6_6881	1	1.61	4778.7	0.20386
denseSNP6_6882	2	1255.68	3523.1	<2e-16
denseSNP6_6881:denseSNP6_6882	1	5.02	3518.0	0.02508

Table 2.1: Anova for linear model fitted to first interaction term that was discovered.

A χ^2 test for independence between denseSNP6_6882 and DR gives a p-value of less than 1e-15, so that GLINTERNET has effectively selected an interaction with DR. However, denseSNP6_6881 has little association with loci A and C. The question then arises as to why we did not find the true interactions with DR. To investigate, we fit a linear logistic regression model separately to each of the two true interaction pairs. In both cases, the main effect DR is significant (p-value $< 1e-15$), but the interaction term is not:

	Df	Dev	Resid. Dev	$\mathbb{P}(> \chi^2)$
NULL			4780.4	
SNP16_31	2	3.08	4777.3	0.2147
DR	5	2383.39	2393.9	<2e-16
SNP16_31:DR	10	9.56	2384.3	0.4797
NULL			4780.4	
denseSNP6_3437	2	1.30	4779.1	0.5223
DR	5	2384.18	2394.9	<2e-16
denseSNP6_3437:DR	8	5.88	2389.0	0.6604

Table 2.2: Anova for linear logistic regression done separately on each of the two true interaction terms.

Therefore it is somewhat unsurprising that GLINTERNET did not pick these interactions.

This example also illustrates how the the group-lasso penalty in GLINTERNET helps in discovering interactions (see Section 2.3.3). We mentioned above that denseSNP6_6881:denseSNP6_6882 is significant if fit by itself in a linear logistic model (Table 2.1). But if we now fit this interaction *in the presence* of the two main effects SNP6_305 and denseSNP6_6873, it is *not* significant:

	Df	Dev	Resid. Dev	$\mathbb{P}(> \chi^2)$
NULL			4780.4	
SNP6_305	2	2140.18	2640.2	<2e-16
denseSNP6_6873	2	382.61	2257.6	<2e-16
denseSNP6_6881:denseSNP6_6882	4	3.06	2254.5	0.5473

This suggests that fitting the two main effects fully has explained away most of the effect from the interaction. But because GLINTERNET regularizes the coefficients of these main effects, they are not fully fit, and this allows GLINTERNET to discover the interaction.

The anova analyses above suggest that the true interactions are difficult to find in this GWAS dataset. Despite having to search through a space of about 360 million interaction pairs, GLINTERNET was able to find variables that are strongly associated with the truth. This illustrates the difficulty of the interaction-learning problem: even if the computational challenges are met, the statistical issues are perhaps the dominant factor.

2.8 Algorithm details

We describe the algorithm used in GLINTERNET for solving the group-lasso optimization problem. Since the algorithm applies to the group-lasso in general and not specifically for learning interactions, we will use $\mathbf{Y}$ as before to denote the n-vector of observed responses, but $\mathbf{X} = [\mathbf{X}_1 \quad \mathbf{X}_2 \quad \ldots \quad \mathbf{X}_p]$ will now denote a generic feature matrix whose columns fall into p groups.

2.8.1 Defining the group penalties γ

Recall that the group-lasso solves the optimization problem

$$\operatorname{argmin}_\beta \mathcal{L}(\mathbf{Y}, \mathbf{X}; \beta) + \lambda \sum_{i=1}^{p} \gamma_i \|\beta_i\|_2, \tag{2.76}$$

where $\mathcal{L}(\mathbf{Y}, \mathbf{X}; \beta)$ is the negative log-likelihood function. This is given by

$$\mathcal{L}(\mathbf{Y}, \mathbf{X}; \beta) = \frac{1}{2n} \|\mathbf{Y} - \mathbf{X}\beta\|_2^2 \tag{2.77}$$

for squared error loss, and

$$\mathcal{L}(\mathbf{Y}, \mathbf{X}, \beta) = -\frac{1}{n}\left[\mathbf{Y}^T(\mathbf{X}\beta) - \mathbf{1}^T \log(\mathbf{1} + \exp(\mathbf{X}\beta))\right] \tag{2.78}$$

for logistic loss (log and exp are taken component-wise). Each β_i is a vector of coefficients for group i. When each group consists of only one variable, this reduces to the lasso.

The γ_i allow us to penalize some groups more (or less) than others. We want to choose the γ_i so that if the signal were pure noise, then all the groups are equally likely to be nonzero. Because the quantity $\|\mathbf{X}_i^T(\mathbf{Y} - \hat{\mathbf{Y}})\|_2$ determines whether the group $\mathbf{X}_i$ is zero or not (see the KKT conditions (1.5)), we define γ_i via a null model as follows. Let $\epsilon \sim (0, I)$. Then we have

$$\begin{aligned} \gamma_i^2 &= \mathbb{E}\|\mathbf{X}_i^T \epsilon\|_2^2 && (2.79)\\ &= \operatorname{tr} \mathbf{X}_i^T \mathbf{X}_i && (2.80)\\ &= \|\mathbf{X}_i\|_F^2. && (2.81) \end{aligned}$$

Therefore we take $\gamma_i = \|\mathbf{X}_i\|_F$, the Frobenius norm of the matrix $\mathbf{X}_i$. In the case where the $\mathbf{X}_i$ are orthonormal matrices with p_i columns, we recover $\gamma_i = \sqrt{p_i}$, which is the value proposed in [49]. In our case, the indicator matrices for categorical variables all have Frobenius norm equal to $\sqrt{n}$, so we can simply take $\gamma_i = 1$ for all i. The case where continuous variables are present is not as straightforward, but we can normalize all the groups to have Frobenius norm one, which then allows us to take $\gamma_i = 1$ for $i = 1, \ldots, p$.

2.8.2 Fitting the group-lasso

Fast iterative soft thresholding (FISTA) [3] is a popular approach for computing the lasso estimates. This is essentially a first order method with Nesterov style acceleration through the use of a momentum factor. Because the group-lasso can be viewed as a more general version of the lasso, it is unsurprising that FISTA can be adapted for the group-lasso with minimal changes. This gives us important advantages:

1. FISTA is a generalized gradient method, so that there is no Hessian involved

2. virtually no change to the algorithm when going from squared error loss to logistic loss

3. gradient computation and parameter updates can be parallelized

4. can take advantage of adaptive momentum restart heuristics.

Adaptive momentum restart was introduced in [29] as a scheme to counter the "rippling" behaviour often observed with accelerated gradient methods. They demonstrated that adaptively restarting the momentum factor based on a gradient condition can dramatically speed up the convergence rate of FISTA. The intuition is that we should reset the momentum to zero whenever the gradient at the current step and the momentum point in different directions. Because the restart condition only requires a vector multiplication with the gradient (which has already been computed), the added computational cost is negligible. The FISTA algorithm with adaptive restart is given below.

Algorithm 1: FISTA with adaptive restart

```
input  : Initialized parameters $\beta^{(0)}$, feature matrix $\mathbf{X}$, observations $\mathbf{Y}$,
         regularization parameter $\lambda$, step size $s$.
output: $\hat{\beta}$
Initialize $x^{(0)} = \beta^{(0)}$ and $\rho_0 = 1$.
for $k = 0, 1, \ldots,$ do
    $g^{(k)} = -\mathbf{X}^T(\mathbf{Y} - \mathbf{X}\beta^{(k)})$;
    $x^{(k+1)} = \left(\mathbf{1} - \frac{s\lambda}{\|\beta^{(k)} - sg^{(k)}\|_2}\right)_+ \left(\beta^{(k)} - sg^{(k)}\right)$;
    $\rho_k = (\beta^{(k)} - x^{(k+1)})^T(x^{(k+1)} - x^{(k)}) > 0 \ ? \ 1 : \rho_k$;
    $\rho_{k+1} = (1 + \sqrt{1 + 4\rho_k^2})/2$;
    $\beta^{(k+1)} = x^{(k+1)} + \frac{\rho_k - 1}{\rho_{k+1}}(x^{(k+1)} - x^{(k)})$;
end
```

At each iteration, we take a step in the direction of the gradient with step size s. We can get an idea of what s should be by looking at the majorized objective function about a fixed point β_0:

$$M(\beta) = \mathcal{L}(\mathbf{Y}, \mathbf{X}; \beta_0) + (\beta - \beta_0)^T g(\beta_0) + \frac{1}{2s}\|\beta - \beta_0\|_2^2 + \lambda \sum_{i=1}^{p} \|\beta_i\|_2. \tag{2.82}$$

Here, $g(\beta_0)$ is the gradient of the negative log-likelihood $\mathcal{L}(\mathbf{Y}, \mathbf{X}; \beta)$ evaluated at β_0. Majorization-minimization schemes for convex optimization choose s sufficiently small so that the LHS of (2.82) is upper bounded by the RHS. One strategy is to start with a large step size, then backtrack until this condition is satisfied. We use an approach that is mentioned in [4] that adaptively initializes the step size with

$$s = \frac{\|\beta^{(k)} - \beta^{(k-1)}\|_2}{\|g_k - g_{k-1}\|_2}. \tag{2.83}$$

We then backtrack from this initialized value if necessary by multiplying s with some $0 < \alpha < 1$. The interested reader may refer to [37] for more details about majorization-minimization schemes for the group-lasso.

2.9 Discussion

We introduced GLINTERNET, a method for learning linear interaction models that satisfy strong hierarchy. We demonstrated that the method is comparable with past approaches, but has the added advantage of being able to accommodate both categorical and continuous variables on a larger scale. We illustrated the method with several examples using real and simulated data, and also showed that GLINTERNET can be applied to genome wide association studies.

GLINTERNET is available on CRAN as a package for the statistical software R.

Chapter 3

EEG source estimation

We now turn to our second application: EEG source estimation via the group-lasso for matrix-valued features. We begin by giving an overview of the problem, past work, and why current methods are inadequate for our purposes.

3.1 Introduction

Non-invasive recordings of human brain activity through electroencephalography (EEG) or magnetoencelphalography (MEG) are of value for both basic science and clinical applications in sensory, cognitive, and affective neuroscience. These methods provide high-temporal resolution measures of neural activity. When combined with inverse modeling techniques, they also provide information about the underlying distribution of neural activity.

3.1.1 Past approaches

The first approach to electromagnetic source localization involved fitting of a single equivalent current dipole to scalp EEG measurements ([34], [44]). Starting in the 1990's, distributed inverse solutions based on the minimum L_2 norm (also known as ridge regression) approach began to appear ([9], [18], [31], [46]). These methods model the underlying source distribution as a large set of elementary currents either distributed throughout the intra-cranial volume or constrained to gray matter. Because the distributed inverse problem is heavily under-determined, there are infinitely many solutions that will recreate the observed signal perfectly. Regularized methods are able to circumvent this problem by penalizing the estimated coefficients, so that one obtains not just a unique source solution, but one that is also more sensible. The L_2 penalty is based on source power: many weakly activated sources are preferred over fewer but stronger sources [39]. Because of this L_2 minimum norm solutions are blurry and contain inverted sign “ghost sources” [17].

Following this early work, alternative penalty functions with L_p norms where $p < 2$ have been applied to the EEG/MEG inverse problem ([15], [26], [43]). A L_1 (lasso) penalty [40] results in estimates of the sources where only a small number of the vertices are nonzero. This has the advantage of being able to produce estimates that are highly localized. However, these approaches can be unstable, and this has limited their wide-spread application: their susceptibility to noise and independent estimation at each time-point causes the highly focal recovered sources to shift unpredictably from locus to locus over time.

Spatial smoothing can alleviate the instability of L_1-penalized methods, but at the expense of the focality of the source estimate. Alternatively, temporal constraints can be imposed to promote smoothness without sacrificing focality ([13], [16], [30], [32], [47], [48]).

A more recent development is to use an elastic-net type of penalty ([51], [25]). These penalties employ a combination of L_1 and L_2 penalties to reap the benefits that each has to offer. They retain the sparsity of the recovered sources that a pure L_1 penalty provides, while the L_2 penalty serves as a smoother that takes care of the instabilities in the L_1 solution. The method we propose in this thesis builds on the elastic-net approach by extending it to *groups* of vertices on a cortical surface mesh: we are able to obtain sparsity on the group level while maintaining smoothness.

3.1.2 Assimilating information from multiple subjects

The approaches mentioned above have largely been applied in the context of single-subject source recovery. One reason is that the methods are inherently unable to pool information across multiple subjects. For example, the L_2 minimum norm approach on S subjects decouples into S individual minimum norm problems, each of which can be solved independently of the others. Pooling can be achieved via a post-processing step. For example, one could average the recovered sources across subjects to get a final estimate.

Instabilities in sparse solutions also pose a problem when performing multi-subject analyses in a common anatomical framework: individual, highly sparse activations tend to not overlap in the common space, leading to low levels of statistical significance with the statistical parametric mapping approaches that are used. A previous solution to this problem uses a hierarchical Bayes technique that fits a Gaussian process with a choice of kernel that imposes group structure ([19], [24]). This framework also utilizes a common anatomical space for inversion in which a template cortical surface is fit to the brains of each individual subject [27].

3.1.3 This thesis

We develop a new source inversion procedure that is based on a generalization of the L_1 penalty called the group-lasso [49]. While the L_1 penalty works by setting the estimates for single sources to zero, the group-lasso is able to zero out sources as a group. In other words, if we label some sources as belonging to the same group, then either all the sources in this group will be estimated to be zero, or *all* are nonzero (some can be zero by random chance, but this is rare).

This property of the group-lasso allows us to avoid using a template procedure to provide a common space for inversion. Instead, we provide a principled approach to defining groups. This approach is based on individually measured regions of interest (ROIs) determined by functional MRI (fMRI). In the example presented here, ROIs are determined for the visual system with retinotopic mapping via fMRI ([1], [2], [8]). The visual system contains several topographic maps of the visual field and these maps correspond to distinct visual areas that have different functional properties. The location of these areas is only loosely constrained to sulci and gyri, particularly as moves from V1 and V2, located in and around the calcarine sulcus to higher order, extra-striate areas. Separating the source via inverse methods is particularly difficult in the visual system because the ROIs can be in close spatial proximity and because of the complexities that result from folding and positioning of the surface of the brain with respect to itself and the sensors. In general, these effects cause some regions to be aliased with others in the inverse, effectively competing with each other in claiming responsibility for the signal. There can also be negative correlations within an area due to tissue orientation effects which create cancellation, i.e. instead of region A correctly showing signal x and region B signal 0, A shows signal $x + y$ and B signal y, because the y cancels out.

Because the ROIs provide a functional grouping of the sources in visual activation studies, the group-lasso naturally applies to this setting. The group-lasso will estimate the sources for an entire ROI to be zero or nonzero, and thus pinpoints which areas are responsible for generating the signal. A second advantage of the group-lasso is that it is able to pool information across multiple subjects in a way that improves the source estimates for individual subjects. In this case a group is the union of the vertices in the corresponding ROIs across the subjects. Recall that the same ROI can have different orientations from subject to subject, so that in some subjects, a ROI might have weak explanatory power for the signal (due to cancellation or correlation with other ROIs), but this same ROI could be strong in other subjects. The group-lasso "settles disputes" by giving the responsibility to the region that appears to be strongest in aggregate over all the subjects. This is illustrated in Figure 3.1.

In the cartoon, we illustrate the brains of six subjects and 3 different ROIs are indicated. The strength of the shading indicates the strength of the recovered signal. In reality the pink and the green ROIs are activated. Due to different positioning and aliasing in their separate forward matrices, in some subjects some of this activation is attributed to the blue ROI. In particular, in subject 1 the blue ROI is stronger than the green. The group-lasso ties the corresponding ROIs across subjects together. It decides collectively, for example, that green is on, in which case it will be on in *all*

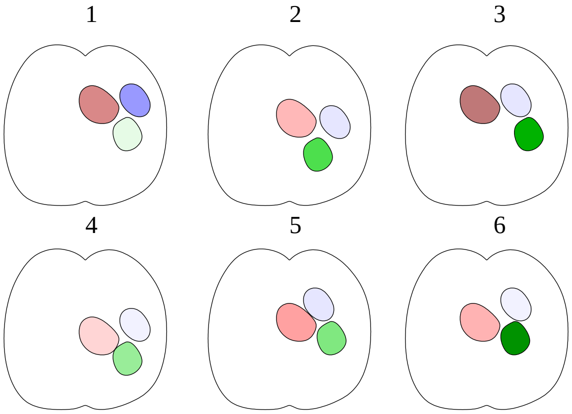


Figure 3.1: Schematic of the group-lasso settling disputes. The true areas are shaded pink and green. The blue region is stronger in subject 1, but pink and green still get chosen over the blue because of their aggregate strength across the other 5 subjects.

subjects (albeit at different strengths in each). In this case, since blue is mostly weak, the blue ROI would be set to zero, and the model would correctly recover the pink and green regions.

We show that inversion with the group-lasso using functional ROIs improves source recovery on simulations above and beyond what can be accomplished with the classical minimum norm for single subjects. We also show that both the minimum norm and group-lasso estimates based on functional ROI constraints improve with increasing numbers of subjects. To the best of our knowledge, this is a novel result. This improvement is more pronounced for the group-lasso than it is for the minimum norm.

We begin with notation in Section 3.2 before describing inversion with the group-lasso in Section 2.3. We evaluate our method and make comparisons with the classical minimum norm solution on simulated data in Section 3.4. We end with a description of the algorithm used to fit the group-lasso in Section 2.8 and a discussion in Section 2.9.

3.2 Notation

We set forth notation that will be used throughout this thesis. We define 18 ROIs per subject (see Section 3.3.1 for details). Let p_i, $i = 1, \ldots, 18$ denote the number of vertices in the i-th ROI, and let $\mathbf{F}_i$, $i = 1, \ldots, 18$ denote the forward matrix for the i-th ROI. We use $\mathbf{Y}$ to represent the $N \times T$ matrix consisting of N observations along T time points, and β_i is the $p_i \times T$ matrix of neural activity in the i-th ROI that we wish to recover. The overall forward matrix is denoted by

$$\mathbf{F} = [\mathbf{F}_1, \mathbf{F}_2, \ldots, \mathbf{F}_{18}], \tag{3.1}$$

and

$$\beta = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_{18} \end{bmatrix} \tag{3.2}$$

is the overall matrix of neural activity. When referring to multiple subjects, we use superscripts to index the subject, so that $\mathbf{F}_i^k$ is the $N \times p_i^k$ forward matrix for subject k's i-th ROI, and similarly for the overall forwards $\mathbf{F}^k$. Note that $\mathbf{F}_i^k$ and $\mathbf{F}_i^l$ can have different numbers of columns, and that there is in general no correspondence between the individual elements of β_i^k and β_i^l.

The forward model that relates the neural activity to the sensor observations can now be expressed as

$$\begin{aligned} \mathbf{Y} &= \mathbf{F}\beta + \epsilon && (3.3) \\ &= \sum_{i=1}^{18} \mathbf{F}_i \beta_i + \epsilon, && (3.4) \end{aligned}$$

where ϵ is a noise term, typically assumed to be distributed as $N(0, \sigma^2 \mathbf{I})$. More complicated models with correlated noise may be more realistic, but are beyond the scope of this work.

3.3 Methodology

3.3.1 Defining regions of interest (ROIs) in the visual cortex

As noted above, grouping of features for group-lasso estimation benefits from a rational basis for defining the groups and here we exploit the multiple retinotopic mappings of the visual field onto the visual cortex to comprise the basis for group formation. For purposes of the present analysis, we defined the detailed 3D shape of each of 18 visual ROIs in 25 participants (V1-L, V1-R, V2v-L, V2v-R, V2d-L, V2d-R, V3v-L, V3v-R, V3d-L, V3d-R, V4-L, V4-R, V3A-L, V3A-R, LOC-L, LOC-R, MT-L, MT-R). These definitions are based on high-resolution T1 anatomical scans combined w functional MRI scans (fMRI). Structural and functional MRI scanning was conducted at 3T (Siemens Tim Trio, Erlangen, Germany) using a 12-channel head coil. We acquired a T1-weighted MRI dataset (3-D MP-RAGE sequence, $0.8 \times 0.8 \times 0.8\ mm^3$ and a 3-D T2-weighted dataset (SE sequence at $1 \times 1 \times 1\ mm^3$ resolution) for tissue segmentation and registration with the functional scans. For fMRI, we employed a single-shot, gradient-echo EPI sequence (TR/TE = 2000/28 ms, flip angle 80, 126 volumes per run) with a vertex size of $1.7 \times 1.7 \times 2\ mm^3$ (128×128 acquisition matrix, 220 mm FOV, bandwidth 1860 Hz/pixel, echo spacing 0.71 ms). We acquired 30 slices without gaps, positioned in the transverse-to-coronal plane approximately parallel to the corpus callosum and covering the whole cerebrum. Once per session, a 2-D SE T1-weighted volume was acquired with the same slice specifications as the functional series in order to facilitate registration of the fMRI data to the anatomical scan.

The FreeSurfer software package (http://surfer.nmr.mgh.harvard.edu) was used to perform gray and white matter segmentation to define a cortical surface mesh with accurate surface normals. The FreeSurfer package extracts both gray/white and gray/cerebrospinal fluid (CSF) boundaries, but these surfaces can have different surface orientations. In particular, the gray/white boundary has sharp gyri (the curvature changes rapidly) and smooth sulci (slowly changing surface curvature), while the gray/CSF boundary is the inverse, with smooth gyri and sharp sulci. We created a new surface that had a similar curvature for both gyri and sulci, avoiding these curvature discontinuities. The new surface generated by interpolating a position that was midway between the gray/white surface and the gray/CSF surface.

The highest accuracy for source-imaging is obtained when there is an accurate model that connects activity at each location on the surface of cortex with how it will be measured at the scalp. To generate realistic scalp topographies, we made separate forward models for each participant in the study using the Boundary Element Method (BEM) with conductivity models that were derived from the T1 and T2 weighted MRI scans of each observer. The FSL toolbox (http://www.fmrib.ox.ac.uk/fsl/) was also used to segment contiguous volume regions for the scalp, outer skull, and inner skull and to convert these MRI volumes into inner skull, outer skull, and scalp surfaces [38].

The general procedures for the scans used to define the visual areas (head stabilization, visual display system, etc) are standard and have been described in detail elsewhere [6]. Retinotopic field mapping defined ROIs for visual cortical areas V1, V2v, V2d, V3v, V3d, V3A, and V4 in each hemisphere ([42], [45]). ROIs corresponding to hMT+ were identified using low contrast motion stimuli similar to those described in [20]. In this study, the fMRI data was used purely to define ROIs for the EEG analysis.

3.3.2 Collaborative effect from multiple subjects

The "all zero or all nonzero" property of the group-lasso estimates allows the group-lasso to pool information across multiple subjects; see Section 3.1.3 for a graphical illustration of this. We thus expect the results of inversion to improve with the number of subjects. One way to make use of the data from multiple subjects is to build a large forward matrix by stacking the individual matrices from each subject, and similarly for the observations. We do not do this because accounting for the different ROI sizes across subjects can get messy, and we also want to impose both spatial (across vertices) and temporal smoothness in the recovered activity. The dimension reduction that results from smoothing also leads to computational speedups.

Because column c of a subject's forward matrix measures the contribution of vertex c to each of the N sensors, we expect neighboring vertices to have roughly the same contribution, that is, the contributions should vary smoothly as we travese the vertices in a ROI. We thus expect the forward matrices $\mathbf{F}_i$ to be low rank where most of the variation can be captured by the top few principal components. We use 5 components because the orientation of a ROI can be parametrized with 3 spatial coordinates along with 2 rotation angles, and this seems to work well in our experiments. This method of spatial smoothing respects the borders of the functional areas: smoothing does not occur across areas that may differ in their functional specificity, as might happen with a purely spatial smoothing such as that used in LORETA [31].

Recall that $\mathbf{F}_i^k$ denotes the $N \times p_i^k$ forward matrix of subject k that corresponds to ROI i. Let $\mathbf{P}_i^k$ denote the $p_i^k \times 5$ matrix consisting of the first 5 right singular vectors of the centered $\mathbf{F}_i^k$. The columns of $\mathbf{P}_i^k$ are a smooth basis across the space of vertices, and we can impose spatial smoothness on the recovered activity by constraining β_i^k to this basis:

$$\beta_i^k = \mathbf{P}_i^k {\beta'}_i^k. \tag{3.5}$$

It follows that each ${\beta'}_i^k$ is a $5 \times T$ matrix. The observed signal contribution from ROI i in subject k can then be written as

$$\begin{aligned} \mathbf{F}_i^k \beta_i^k &= \mathbf{F}_i^k \mathbf{P}_i^k {\beta'}_i^k && (3.6) \\ &= \mathbf{X}_i^k {\beta'}_i^k, && (3.7) \end{aligned}$$

where

$$\mathbf{X}_i^k = \mathbf{F}_i^k \mathbf{P}_i^k \tag{3.8}$$

is the $N \times 5$ matrix consisting of the first 5 principal components of $\mathbf{F}_i^k$. We call $\mathbf{X}_i^k$ the *derived forward matrix* for ROI i in subject k. The overall derived forward matrix for S subjects can then be constructed by

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1^1 & & & & \cdots & \mathbf{X}_{18}^1 & & & \\ & \mathbf{X}_1^2 & & & \cdots & & \mathbf{X}_{18}^2 & & \\ & & \ddots & & \cdots & & & \ddots & \\ & & & \mathbf{X}_1^S & \cdots & & & & \mathbf{X}_{18}^S \end{bmatrix}. \tag{3.9}$$

We can write $\mathbf{X}$ in more compact form (recall that subscripts index ROIs and superscripts index subjects) by

$$\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_{18}], \tag{3.10}$$

where

$$\mathbf{X}_i = \begin{bmatrix} \mathbf{X}_i^1 & & & \\ & \mathbf{X}_i^2 & & \\ & & \ddots & \\ & & & \mathbf{X}_i^S \end{bmatrix}. \tag{3.11}$$

Similarly, we construct the spatially smoothed activity for S subjects for ROI i by writing

$$\beta_i' = \begin{bmatrix} {\beta'}_i^1 \\ {\beta'}_i^2 \\ \vdots \\ {\beta'}_i^S \end{bmatrix}. \tag{3.12}$$

Combining observations from various subjects is more straightforward, and can be done by simply stacking the observations:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y}^1 \\ \mathbf{Y}^2 \\ \vdots \\ \mathbf{Y}^S \end{bmatrix}. \tag{3.13}$$

The group-lasso objective for the derived forward matrices $\mathbf{X}_i$ and the spatially smoothed activity β' can now be written as

$$\text{argmin}_{\mu,\beta'} \frac{1}{2} \left\| \mathbf{Y} - \mathbf{1}\mu^t - \sum_{i=1}^{18} \mathbf{X}_i \beta'_i \right\|_F^2 + \lambda \sum_{i=1}^{18} \gamma_i \|\beta'_i\|_F + \alpha \|\beta'\|_F^2. \tag{3.14}$$

The group-lasso has an important advantage that none of the previously mentioned methods have. Because a group now consists of a single ROI across multiple subjects, there is a collaborative effect in that as long as a ROI has a strong signal in enough subjects, we will estimate that ROI to be nonzero even in those subjects where that ROI is not quite lighting up. Consider subject 1 in Figure 3.1. The blue region (wrong) appears to be stronger than the green region (correct). This could be due to the blue region being highly correlated with the green region, noise in the data, or some other region that is cancelling out the signal from the green region. In any case, the blue region is competing with the green region for attention, and if we tried to recover the activity with subject 1 alone, the group-lasso is likely to pick the blue region over the green region, resulting in a mistake. But if we fit the group-lasso on all 6 subjects, the other 5 subjects would clearly vote in favor of the green over the blue, and as a result, we will select the green region for subject 1. And because the blue region is zeroed out in the other 5 subjects, it will also be zeroed out in subject 1, i.e. there is no competition from the blue region, and the green region can be fully fit to the signal. This is to be constrasted with a method like the minimum norm or elastic-net, where both methods would be happy with dividing the signal between the green and blue regions in order to get the best fit, so that both would be nonzero.

The minimum norm and elastic-net are inherently unable to pool information across subjects. Any such pooling has to be done manually as a postprocessing step, such as averaging the estimated sources over the multiple subjects. Naturally, this can be done for the group-lasso as well. We expect this pooling effect to be stronger as the number of subjects increases, and we show in Section 3.4 that this is indeed the case.

3.3.3 Imposing temporal smoothness

In the spirit of the previous section, it is reasonable to assume that the neural activity also varies smoothly over time, and we can impose temporal smoothness in the estimated source by finding a suitable basis for the time component. The right singular vectors of $\mathbf{Y}$ are a natural basis for the temporal component:

$$\mathbf{Y}_{N\times T} = \mathbf{U}_{N\times N}\mathbf{D}_{N\times T}\mathbf{V}^t_{T\times T}. \tag{3.15}$$

The singular value decomposition is also used in [13] to obtain the principal directions along the time axis, but there they use 5 singular vectors. We fix the dimension d of this basis by taking as many singular vectors as we need to explain 99% of the variance of $\mathbf{Y}$. In particular, this is given by

$$d = \text{argmin}_k \left\{ k : \frac{\sum_{i=1}^k d_{ii}^2}{\sum_{i=1}^N d_{ii}^2} \geq 0.99 \right\}. \tag{3.16}$$

In our experiments, this number is typically 2. Let $\mathbf{V}_d$ be the matrix consisting of the first d columns of $\mathbf{V}$. We restrict β' to the space spanned by $\mathbf{V}_d$ by setting

$$\beta' = \tilde{\beta}\mathbf{V}_d^t. \tag{3.17}$$

Applying this restriction to (3.14) gives

$$\text{argmin}_{\mu,\tilde{\beta}} \frac{1}{2}\left\| \mathbf{Y} - \mathbf{1}\mu^t - \sum_{i=1}^{18} \mathbf{X}_i\tilde{\beta}_i\mathbf{V}_d^t \right\|_F^2 + \lambda\sum_{i=1}^{18}\gamma_i\|\tilde{\beta}_i\mathbf{V}_d^t\|_F + \alpha\|\tilde{\beta}\mathbf{V}_d^t\|_F^2. \tag{3.18}$$

Because $\mathbf{V}_d$ is orthonormal, this is equivalent to

$$\operatorname{argmin}_{\mu,\tilde{\beta}} \frac{1}{2} \left\| \tilde{\mathbf{Y}} - \mathbf{1}\mu^t - \sum_{i=1}^{18} \mathbf{X}_i \tilde{\beta}_i \right\|_F^2 + \lambda \sum_{i=1}^{18} \gamma_i \|\tilde{\beta}_i\|_F + \alpha \|\tilde{\beta}\|_F^2, \tag{3.19}$$

where $\tilde{\mathbf{Y}} = \mathbf{Y}\mathbf{V}_d$ is the $N \times d$ matrix of temporally smoothed observations.

After spatial and temporal smoothing, we have, for S subjects, the $NS \times d$ matrix $\tilde{\mathbf{Y}}$ of smoothed observations, the $NS \times (S \cdot 18 \cdot 5)$ derived forward matrix $\mathbf{X}$, and the $(S \cdot 18 \cdot 5) \times d$ matrix $\tilde{\beta}$ of smoothed activity that we need to estimate. This is achieved by solving (3.19).

3.3.4 Recovering the activity in the original space

We transform the estimated $\hat{\tilde{\beta}}$ back to the original space by reversing the temporal smoothing and dimension reduction operations. We illustrate for a single subject. Let

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_1 & & \\ & \ddots & \\ & & \mathbf{P}_{18} \end{bmatrix} \tag{3.20}$$

denote the block diagonal matrix consisting of the P_i's in (3.5). From (3.5), it is clear that reversing the spatial smoothing can be done by left-multiplying our solution $\hat{\tilde{\beta}}$ by $\mathbf{P}$. Similarly, (3.17) shows that right-multiplying by $\mathbf{V}_d^t$ reverses the temporal smoothing. To summarize, our estimate of the source activity in the original space is given by

$$\hat{\beta} = \mathbf{P}\hat{\tilde{\beta}}\mathbf{V}_d^t. \tag{3.21}$$

3.3.5 Model selection

Generalized cross validation (GCV) is one method of model selection that is intuitively simple and widely used. Let $\mathbf{Y}$ be the $N \times T$-matrix of observations, and $\hat{\mathbf{Y}}$ the fitted values. The GCV error for this fit is given by

$$\frac{\frac{1}{NT}\|\mathbf{Y} - \hat{\mathbf{Y}}\|_F^2}{(1 - \frac{df(\hat{\mathbf{Y}})}{NT})^2}, \tag{3.22}$$

where $df(\hat{\mathbf{Y}})$ is the degrees of freedom for $\hat{\mathbf{Y}}$. Fitting the group-lasso along a grid of λ values results in a GCV error curve. We then pick the λ that gives the minimum value on this curve. Full details on determining the degrees of freedom $df(\hat{\mathbf{Y}})$ for the group-lasso solutions and selecting the α parameter in (1.8) are given in Appendix A.

3.4 Results

We make comparisons between the group-lasso and minimum norm methods through simulation. We also evaluate the methods on multiple subjects (up to 25), and demonstrate that the effectiveness of the group-lasso increases with the number of subjects. The minimum norm method does not inherently pool information across multiple subjects, but we can average the recovered acitivity across subjects for each ROI as a post-processing step. This ROI-based averaging improves performance for both the group-lasso and the minimum norm. We begin by describing the simulation process.

3.4.1 Experimental setup

We select ROIs V2v-L and V4-R as the ground truth. These areas are separated by about a centimeter, on average, and exhibit considerable cross-talk between their forward vectors. The neural activity in each is generated as follows. For each region we randomly sample a number from $\{1, 1.1, 1.2, \ldots, 1.9, 2, 2.1, \ldots, 9.9, 10\}$, then assign a randomly chosen cluster within that region with this value. This cluster comprises about 30% of the total size of the region. Therefore V2v-L and V4-R each have constant activity in about 30% of their vertices. We then pass the activity through the forward model to obtain the observed time courses $\mathbf{Y}$. We add Gaussian white noise to $\mathbf{Y}$ to obtain a signal to noise ratio of 0.32. In all cases, we take $N = 128$ observations/sensors and $T = 91$ time points.

The methods are evaluated on 3 measures (see [8] for more details):

1. Area under the ROC curve (AUC)
2. Mean squared error (MSE) on the neural activity, given by $\frac{1}{NT}\|\beta - \hat{\beta}\|_F^2$
3. Relative energy, given by the ratio between the normalized energies contained in the estimate of the active sources and the global distribution:

$$\frac{\sum_{i \in \mathcal{A}} E_{est}(i)}{\sum_i E_{est}(i)}, \tag{3.23}$$

   where $\mathcal{A}$ is the set of active vertices in the true neural activity and $E_{est}(i)$ is the energy of the estimated signal at vertex i.

For a single subject, we compute these metrics for each of the T time points, then take the average. For multiple subjects, we compute this time average separately for each subject, and then take the average across all subjects.

3.4.2 A single subject

We illustrate the performance of the group-lasso inversion for a single subject on one instance of simulated data. Because both the group-lasso and minimum norm methods produce a sequence of fits, we can visualize their performance as we move along their solution paths. One way to do this is to plot their performance as a function of variance explained (r^2) on the training data. The r^2 is defined by

$$r^2 = 1 - \frac{\|\mathbf{Y} - \hat{\mathbf{Y}}\|_F^2}{\|\mathbf{Y} - \bar{\mathbf{Y}}\|_F^2} \tag{3.24}$$

where $\hat{\mathbf{Y}}$ is the fit and $\bar{\mathbf{Y}}$ is the matrix whose i-th column is the mean of the i-th column of $\mathbf{Y}$. We plot the 3 metrics as a function of r^2 in Figure 3.2.

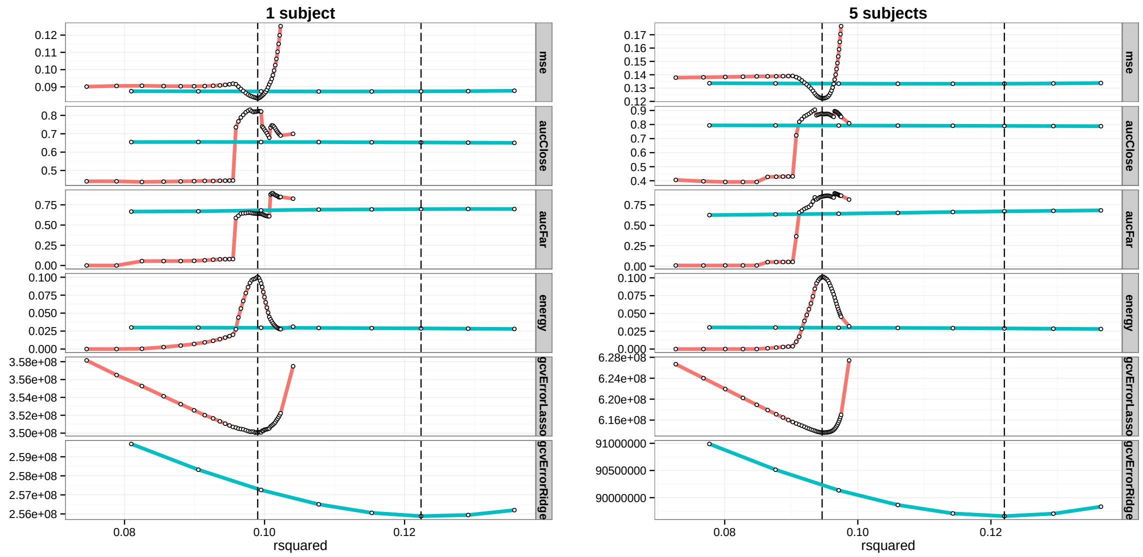


Figure 3.2: Performance of the group-lasso and minimum norm on one instance of simulated data for a one and five subjects. Vertical lines correspond to the solutions chosen by optimizing the GCV error curve for each method. Note that GCV errors are computed with the observations (and their fitted values), while MSE measures goodness of fit to the activity, which explains the difference in scale between the two. **Red:** group-lasso. **Blue:** minimum norm.

We see that the group-lasso outperforms the minimum norm on AUC and relative energy, and is comparable on MSE. This is to be expected because there is no pooling effect for a single subject. The group-lasso, like the minimum norm, is unable to leverage the information from other subjects to effectively distinguish between confounding ROIs.

3.4.3 5 subjects

We selected 5 subjects at random from our database and made the same comparison as in the previous section on a single instance of simulated data. While the results are qualitatively similar, notice that the group-lasso does better than in the single subject case, as expected. This is because it is now able to assimilate information from other subjects to decide if a ROI should be activated or not. The minimum norm solution does not aggregate information across the multiple subjects, so that its performance is similar to the single subject case.

Notice that the group-lasso solution near the end of the path exhibits large MSE in both the 1 and 5 subject cases. This is is likely due to the spatial smoothing and temporal smoothing, but mostly due to the former. To see this, we generate activity β for a single subject, then compute

$$\|\beta - \mathbf{P}\mathbf{P}^T\beta\|_F^2 \tag{3.25}$$

for varying numbers of principal components, and

$$\|\beta - \beta\mathbf{V}_d\mathbf{V}_d^T\|_F^2 \tag{3.26}$$

for varying d ($\mathbf{P}$ and $\mathbf{V}_d$ defined in Sections 3.3.2 and 3.3.3). These computations tell us how we can expect to perform on MSE if we knew the true activity β, but subjected it to the smoothness constraints.

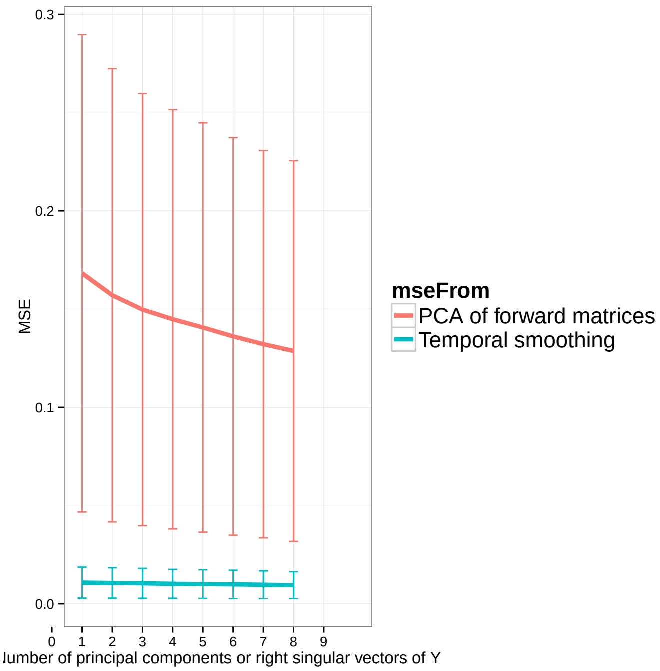


Figure 3.3: MSE from dimension reduction by principal components and temporal smoothing with right singular vectors of $\mathbf{Y}$, averaged over 100 simulations. A large portion of the MSE is due to the dimension reduction from taking the first 5 principal components for each ROI.

The observed MSE is on the order of that seen in Figure 3.2. The rapid increase in MSE is also due to an increase in variance as we decrease the amount of regularization. For a single subject, there are Nd observations and $18 \cdot 5 \cdot d$ parameters (see the end of Section 3.3.3), so that as $\lambda \downarrow 0$, we approach a near-saturated fit.

3.4.4 Better performance with more subjects

We next investigate how the performance of the group-lasso scales with the number of subjects. We take 1, 2, 4, 8, 16, and 25 subjects, and for each situation, we fit the group-lasso and minimum norm methods on 20 different instances of simulated data. We then compute the average performance across the 20 simulations.

The metrics, along with standard error bars, are plotted as a function of the number of subjects in Figure 3.4.

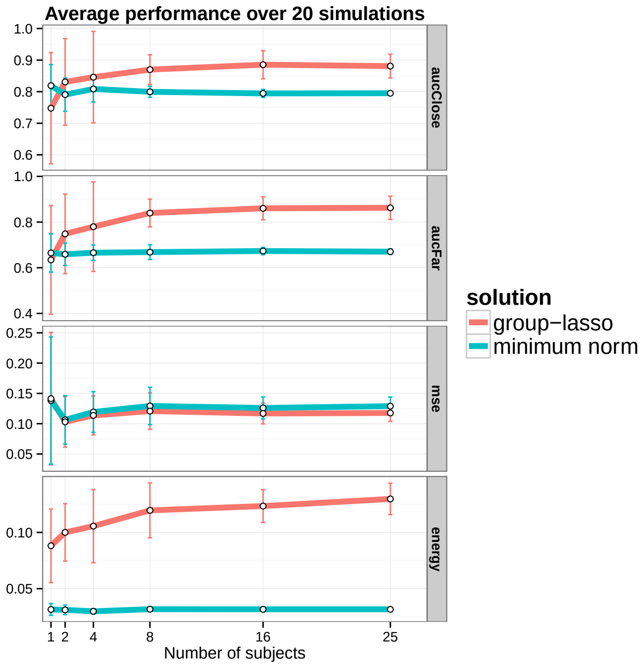


Figure 3.4: Performance of the group-lasso and minimum norm as a function of the number of subjects. Plots are of averages from 20 simulations. Vertical lines are standard error bars.

As before, the minimum norm solution does not pool information across multiple subjects, so that the performance stays flat despite having more subjects. The group-lasso clearly benefits from having more subjects, but this benefit tapers off after about 8 subjects.

We mentioned in Section 3.1.3 that both methods benefit from a post-processing step that averages the estimated activity within a ROI across multiple subjects. We do this on the same data instances generated in the above simulation and plot the AUC as a function of the number of subjects in Figure 3.5.

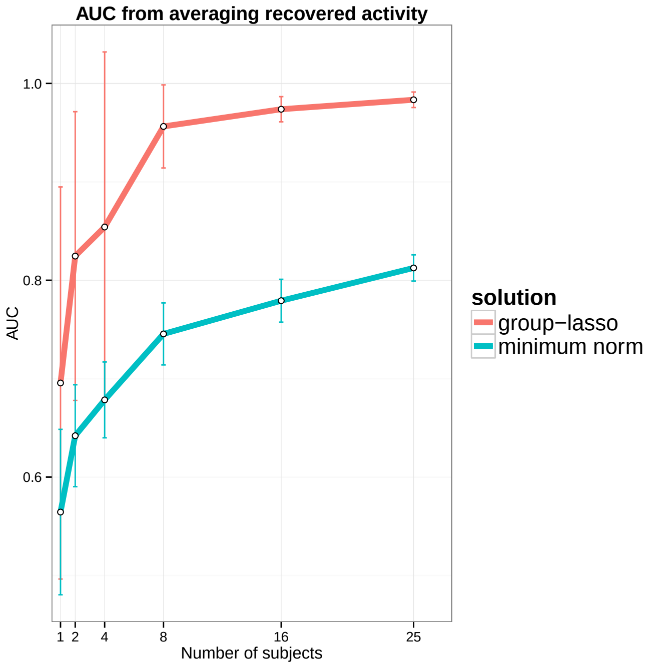


Figure 3.5: AUC obtained after post-processing the recovered activity by averaging across subjects. Plots are of average values over the same 20 data instances from before, along with standard error bars. Notice that the group-lasso with 2 subjects often outperforms the minimum norm with 25 subjects.

Both methods get a substantial boost from ROI-based averaging across subjects, and performance improves as the number of subjects is increased. This is a novel result, and is distinct from with the effect shown in Figure 3.4. There, the performance gain is due to the "majority vote" mechanism of the group-lasso as illustrated in Figure 3.1. Post-processing the recovered activity by ROI averaging serves to further reduce the variance in the estimates, thus resulting in a higher AUC for not just the group-lasso, but also for the minimum norm.

3.5 Algorithm details

We describe the algorithm used to obtain the solutions to (1.8). Cyclic group-wise coordinate descent works well when we have a small number of groups (18 ROIs in our case). The idea is to update the coefficients for a single group while holding the coefficients for all other groups fixed. If we cycle through all the groups repeatedly, we will converge on a solution. Let $\mathbf{Y}$ be the matrix of observations, and let $\mathbf{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_p]$ be the matrix of features that consists of p groups.

3.5.1 Cyclic group-wise coordinate descent

Let λ and α be fixed, and suppose we want to perform the update for group k. Let $\hat{\beta}_1, \ldots, \hat{\beta}_p$ be the current estimates with current residual given by $\mathbf{r} = \mathbf{Y} - \sum_{i=1}^{p} \mathbf{X}_i \beta_i$, and let $\mathbf{r}_k = \mathbf{r} + \mathbf{X}_k \beta_k$ be the partial residual for group k. If $\hat{\beta}_k = 0$, the gradient equation for $\hat{\beta}_k$ is given by

$$\mathbf{X}_k^T \mathbf{r}_k = \lambda \gamma_k \mathbf{s}_k, \tag{3.27}$$

where $\mathbf{s}_k \in \{\mathbf{z} : \|\mathbf{z}\|_F \leq 1\}$. It follows that $\hat{\beta}_k = 0$ if $\|\mathbf{X}_k^T \mathbf{r}_k\|_F < \lambda\gamma_k$. If $\hat{\beta}_k \neq 0$, the gradient condition for optimality is given by

$$\left(\mathbf{X}_k^T \mathbf{X}_k + \left(\frac{\lambda \gamma_k}{\|\hat{\beta}_k\|_F} + 2\alpha\right)\mathbf{I} \right) \hat{\beta}_k = \mathbf{X}_k^T \mathbf{r}_k \tag{3.28}$$

We can solve for $\hat{\beta}_k$ by first solving for $\|\hat{\beta}_k\|_F$ and then plugging the result into (3.28).

Take the singular value decomposition $\mathbf{X}_k = \mathbf{U}_k \mathbf{D}_k \mathbf{V}_k^T$ and rewrite (3.28) as

$$\left[\mathbf{D}_k^2 \|\hat{\beta}_k\|_F + (\lambda\gamma_k + 2\alpha\|\hat{\beta}_k\|_F)\mathbf{I} \right]^{-1} \mathbf{D}_k \mathbf{U}_k^T \mathbf{r}_k = \mathbf{V}_k^T \frac{\hat{\beta}_k}{\|\hat{\beta}_k\|_F}. \tag{3.29}$$

Take the Frobenius norm on both sides to obtain

$$\left\| \left[\mathbf{D}_k^2 \|\hat{\beta}_k\|_F + (\lambda\gamma_k + 2\alpha\|\hat{\beta}_k\|_F)\mathbf{I} \right]^{-1} \mathbf{D}_k \mathbf{U}_k^T \mathbf{r}_k \right\|_F^2 = 1. \tag{3.30}$$

Let $f(\theta) = \left\| [\mathbf{D}_k^2 \theta + (\lambda\gamma_k + 2\alpha\theta)\mathbf{I}]^{-1} \mathbf{D}_k \mathbf{U}_k^T \mathbf{r}_k \right\|_F^2 - 1$. To find $\|\hat{\beta}_k\|_F$, we need only find

θ_0 such that $f(\theta_0) = 0$. We do this with Newton-Rhapson by reiterating

$$\theta \longleftarrow \theta - \eta \frac{f(\theta)}{f'(\theta)}, \tag{3.31}$$

where η is the step size and

$$f'(\theta) = -2 \left\| \left[\mathbf{D}_k^2 \theta + (\lambda\gamma_k + 2\alpha\theta)\mathbf{I} \right]^{-\frac{3}{2}} (\mathbf{D}_k^2 + 2\alpha\mathbf{I})^{\frac{1}{2}} \mathbf{D}_k \mathbf{U}_k^T \mathbf{r}_k \right\|_F^2 . \tag{3.32}$$

In our experience, $f(\theta)$ tends to be quite linear around θ_0, so that very few Newton iterations are required for convergence. Having obtained $\hat{\theta}_0$, we update $\hat{\beta}_k$ with

$$\hat{\beta}_k \longleftarrow \left(\mathbf{X}_k^T \mathbf{X}_k + (\frac{\lambda\gamma_k}{\hat{\theta}_0} + 2\alpha)\mathbf{I} \right)^{-1} \mathbf{X}_k^T \mathbf{r}_k. \tag{3.33}$$

We now cycle through all the groups until convergence. The full algorithm is presented in Algorithm 2. In practice, because we are fitting the group-lasso along a sequence of λ, we will initialize $\hat{\beta}_1, \ldots, \hat{\beta}_p$ with the estimates from the previous λ in the sequence. These "warm starts" give a significant speed advantage in our experience.

3.5.2 Determining the group penalty modifiers γ_i

The γ_i in (1.8) allow us to have different penalties for different groups. This is useful because a larger group can be more likely to have a stronger correlation with the response than a small group, just by random chance. Having different penalties thus allows us to put different-sized groups on the same scale.

Recall that $\hat{\beta}_k = 0$ if the following gradient condition is met:

$$\|\mathbf{X}_k^T \mathbf{r}_k\|_F < \lambda\gamma_k. \tag{3.34}$$

It follows that we can determine an appropriate group penalty modifier by computing the expected value of the LHS if the signal were pure noise. Let $\epsilon \sim (0, \mathbf{I})$. Then we

Algorithm 2: Cyclic group-wise coordinate descent

```
input  : $\mathbf{Y}$, $\mathbf{X}_1, \ldots, \mathbf{X}_p$, $\lambda, \gamma_1, \ldots, \gamma_p, \alpha$
output: $\hat{\beta}_1, \ldots, \hat{\beta}_p$
Initialize $\mathbf{r} = \mathbf{Y} - \bar{\mathbf{Y}}$, $\hat{\beta}_1 = 0, \ldots, \hat{\beta}_p = 0$. Let $\mathbf{X}_k = \mathbf{U}_k \mathbf{D}_k \mathbf{V}_k^T$ be the singular decomposition of $\mathbf{X}_k$.
Iterate till convergence:
for $k \leftarrow 1$ to $p$ do
    $\mathbf{r}_k = \mathbf{r} + \mathbf{X}_k \hat{\beta}_k$;
    if $\|\mathbf{X}_k^T \mathbf{r}_k\|_F < \lambda\gamma_k$ then
        $\hat{\beta}_k \longleftarrow 0$;
    end
    else
        $\hat{\beta}_k \longleftarrow \left[\mathbf{X}_k^T \mathbf{X}_k + (\frac{\lambda\gamma_k}{\theta} + 2\alpha)\mathbf{I}\right]^{-1} \mathbf{X}_k^T \mathbf{r}_k$;
        where $\theta$ is the root of $\| \left[\mathbf{D}_k^2 \theta + (\lambda\gamma_k + 2\alpha)\mathbf{I}\right]^{-1} \mathbf{D}_k \mathbf{U}_k^T \mathbf{r}_k \|_F = 1$
    end
    $\mathbf{r} \longleftarrow \mathbf{r}_k - \mathbf{X}_k \hat{\beta}_k$;
end
return $\hat{\beta}_1, \ldots, \hat{\beta}_p$
```

have

$$\begin{aligned}
\gamma_k^2 &= \mathbb{E}\|\mathbf{X}_k^T\epsilon\|_F^2 && (3.35)\\
&= \mathbb{E}\,\mathrm{tr}\,\epsilon^T\mathbf{X}_k\mathbf{X}_k^T\epsilon && (3.36)\\
&= \mathrm{tr}\,\mathbb{E}\epsilon^T\mathbf{X}_k\mathbf{X}_k^T\epsilon && (3.37)\\
&= \mathrm{tr}\,\mathbf{X}_k^T\mathbf{X}_k && (3.38)\\
&= \|\mathbf{X}_k\|_F^2. && (3.39)
\end{aligned}$$

Therefore we take $\gamma_k = \|\mathbf{X}_k\|_F$, the Frobenius norm of $\mathbf{X}_k$. Note that if $\mathbf{X}_k$ is orthonormal, then $\gamma_k = \sqrt{p_k}$, which is the penalty modifier proposed in [49].

3.6 Discussion

This work introduces a new approach toward EEG source estimation in the visual cortex using the group-lasso. Because the group-lasso selects variables in a group-wise manner, we do not require a template procedure to provide a common space for source inversion. Instead, we use ROIs that are determined by fMRI to define the groups. We show how to combine data from multiple subjects while imposing spatial and temporal smoothness on the recovered activity. The “pooling effect” of the group-lasso suggests that its performance should improve with the number of subjects, and we verified this with simulated experiments. In particular, the performance of the group-lasso is comparable to that of the traditional minimum norm solution, but as the number of subjects increases, there is a significant performance boost for the group-lasso over the minimum norm.

We also show that while the minimum norm does not inherently pool information across multiple subjects, it still benefits from a post-processing step in which the recovered activity is averaged across subjects. This, to the best of our knowledge, is a novel result. Of course, this averaging can also be applied to the group-lasso solution, which leads to a further performance increase beyond that already obtained from its inherent pooling effect.

The effectiveness of the group-lasso can lead to other interesting possibilities. The overlapped group-lasso is a special case of the group-lasso in which a variable can show up in more than one group. It follows that the overlapped group-lasso might be a good choice for source inversion in cases where the ROIs have overlaps.

Another aspect of ROI-wise source inversion that we have not explored is sparsity within a ROI. It is possible that the source activity is only present in some fraction of the ROI, so that a solution that is sparse within a ROI is desirable. If this is the case, an additional L_1 penalty of the form $\|\beta\|_1$ can be added to the group-lasso penalty to impose sparsity. This results in what is known as the sparse-group lasso (see [37] for details). This, along with the overlapped group-lasso, will be addressed in future work.

Chapter 4

Conclusion

We applied the group-lasso to two problems from different domains. In the interaction learning problem, an overlapped group-lasso penalty allowed us to obtain estimates that satisfy strong hierarchy: if an interaction is estimated to be nonzero, then its associated main effects will also be estimated to be nonzero. We showed that the resulting complicated constrained optimization problem can be tackled by solving an *unconstrained* problem that gives equivalent solutions. This is the basis for GLINTERNET, which will be available as a R package on CRAN.

The availability of regions of interest in the visual cortex was a key ingredient in our solution. Because these ROIs define a functional grouping of the vertices in the visual cortex, it fitted naturally into the group-lasso framework. We showed that the group-lasso benefits from having data from multiple subjects, resulting in better performance over the standard minimum norm solution. We also demonstrated that both the group-lasso and the minimum norm benefit from a post-processing step in which the recovered activity is averaged (within ROI) across multiple subjects.

Appendix A

Model selection details

To the best of our knowledge, there is no analytic form for the degrees of freedom for the group-lasso. We use the results in [22] as a heuristic for estimating the degree of freedom. Suppose there are G groups of variables with sizes $p_1, \ldots, p_G$. Let $\hat{\beta}^0$ denote the full ordinary least squares fit, $\hat{\beta}_i$ the group-lasso estimate for group i, and $\hat{\mathbf{Y}}$ the fit. Then if the feature matrix $\mathbf{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_G]$ is orthonormal, an unbiased estimate of the degrees of freedom is given by

$$df(\hat{\mathbf{Y}}) = \sum_{i=1}^{G} \mathbf{1}(\|\hat{\beta}_i\|_2 > 0) + \sum_{i=1}^{G} (p_i - 1) \frac{\|\hat{\beta}_i\|_2}{\|\hat{\beta}_i^0\|_2}. \tag{A.1}$$

Because $\mathbf{X}$ is not orthonormal in our application (see Section 3.3.2), we do not expect this formula to hold exactly. Simulations show that the estimate given by (A.1) can be biased low, which results in an overly optimistic GCV error rate. The bias is due to large variance in $\hat{\beta}^0$, making the contribution from the $\frac{\|\hat{\beta}_i\|_2}{\|\hat{\beta}_i^0\|_2}$ term negligible. Adding a ridge penalty to $\hat{\beta}^0$ largely eliminates this problem. To verify this, we generate data according to the following setup.

Let $\mathbf{Y}_{fixed}$ be a fixed $N \times T$ matrix of observations. If $\mathbf{Y} = \mathbf{Y}_{fixed} + \epsilon$ with

$\epsilon \sim (0, \sigma^2\mathbf{I})$, the degrees of freedom of a fit $\hat{\mathbf{Y}}$ is given by [10]

$$\begin{aligned}
\frac{1}{\sigma^2}\sum_{t=1}^{T}\sum_{i=1}^{N}\mathrm{Cov}(\mathbf{Y}_{it}, \hat{\mathbf{Y}}_{it}) &= \frac{1}{\sigma^2}\sum_{t=1}^{T}\mathrm{tr}\,\mathrm{Cov}(\mathbf{Y}_{\cdot t}, \hat{\mathbf{Y}}_{\cdot t}) && \text{(A.2)}\\
&= \frac{1}{\sigma^2}\sum_{t=1}^{T}\mathrm{tr}\,\mathrm{Cov}(\epsilon_{\cdot t}, \hat{\mathbf{Y}}_{\cdot t}) && \text{(A.3)}\\
&= \frac{1}{\sigma^2}\sum_{t=1}^{T}\mathrm{tr}\,\mathbb{E}\epsilon_{\cdot t}\hat{\mathbf{Y}}_{\cdot t}^T && \text{(A.4)}\\
&= \frac{1}{\sigma^2}\sum_{t=1}^{T}\mathbb{E}\,\mathrm{tr}\,\epsilon_{\cdot t}\hat{\mathbf{Y}}_{\cdot t}^T && \text{(A.5)}\\
&= \frac{1}{\sigma^2}\sum_{t=1}^{T}\mathbb{E}\hat{\mathbf{Y}}_{\cdot t}^T\epsilon_{\cdot t} && \text{(A.6)}\\
&= \frac{1}{\sigma^2}\mathbb{E}\,\mathrm{tr}\,\hat{\mathbf{Y}}^T\epsilon && \text{(A.7)}
\end{aligned}$$

It follows that we can estimate the true degrees of freedom by generating noisy observations $\mathbf{Y} = \mathbf{Y}_{fixed} + \epsilon$ and averaging the trace of $\hat{\mathbf{Y}}^T\epsilon$ over many simulations. We can then compare the results with the formula in (A.1). Figure A.1 shows the results from 1000 simulations with $\sigma = 5$.

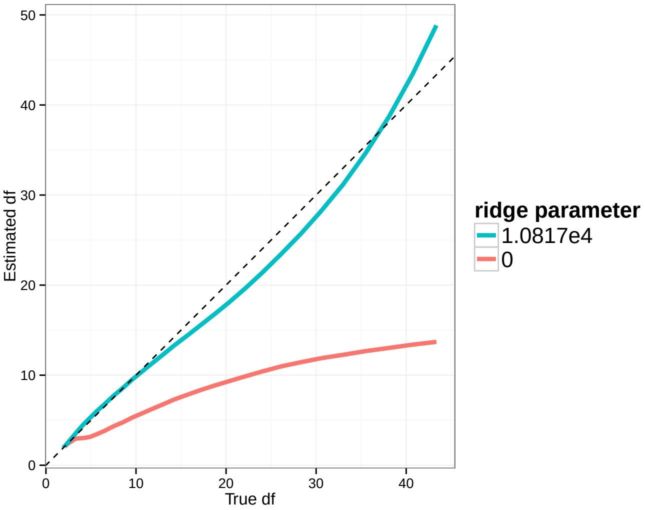


Figure A.1: Estimated degrees of freedom (using (A.1)) vs true df. **Red line:** Using formula (A.1) without any ridge penalty to $\hat{\beta}^0$ results in an estimate that is biased downward. **Blue line:** In our experiments, a ridge penalty of 1.0817×10^4 works well.

Figure A.2 shows that adding a ridge penalty can dramatically reduce the variance in $\hat{\beta}^0$.

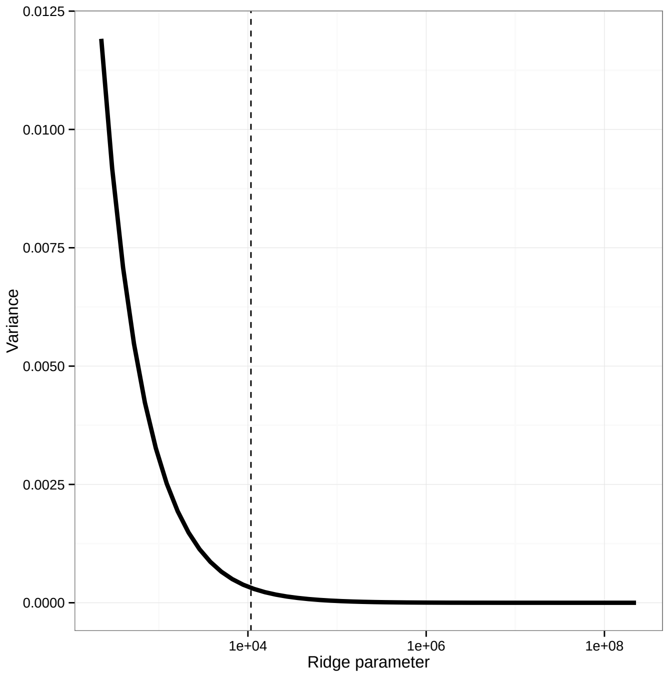


Figure A.2: Variance of $\hat{\beta}^0$ as a function of ridge parameter. Vertical line corresponds to 1.0817×10^4 that is found to work well in our degrees of freedom simulations.

We have seen that adding a ridge penalty to the ordinary least squares fit in the degrees of freedom formula leads to a more stable and accurate estimate. Because the group-lasso estimates converge to the ordinary least squares estimates (for "$p < n$" problems) as $\lambda \downarrow 0$, we use the same amount of ridging by setting $\alpha = 1.0817 \times 10^4$ in (1.8). We use this value from here on.

Bibliography

[1] J.M. Ales and A.M. Norcia. Assessing direction-specific adaptation using the steady-state visual evoked potential: results from eeg source imaging. *Journal of Vision*, 9(7)(8):1–13, 2009.

[2] L. G. Appelbaum, A. R. Wade, V. Y. Vildavski, M. W. Pettet, and A. M. Norcia. Cue-invariant networks for figure and background processing in human visual cortex. *J Neurosci*, 26(45):11695–708, 2006.

[3] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Img. Sci.*, 2(1):183–202, March 2009.

[4] StephenR. Becker, EmmanuelJ. Candes, and MichaelC. Grant. Templates for convex cone problems with applications to sparse signal recovery. *Mathematical Programming Computation*, 3(3):165–218, 2011.

[5] J. Bien et al. A lasso for hierarchical interactions. *The Annals of Statistics*, 2013.

[6] AA Brewer, J Liu, AR Wade, and BA Wandell. Visual field maps and stimulus selectivity in human ventral occipital cortex. *Nat Neurosci*, 8:1102–1109, 2005.

[7] C.C.M. Chen, H. Schwender, J. Keith, R. Nunkesser, K. Mengersen, and P. Macrossan. Methods for identifying snp interactions: A review on variations of logic regression, random forest and bayesian logistic regression. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, 8(6):1580 –1591, nov.-dec. 2011.

[8] B. R. Cottereau, J. M. Ales, and A. M. Norcia. Increasing the accuracy of electromagnetic inverses using functional area source correlation constraints. *Hum Brain Mapp*, 33(11):2694–713, 2012.

[9] A. Dale and M. Sereno. Improved localization of cortical activity by combining eeg and meg with mri cortical surface reconstruction: a linear approach. *J Cogn Neurosci*, 5(162-176), 1993.

[10] Bradley Efron. How Biased is the Apparent Error Rate of a Prediction Rule? *Journal of the American Statistical Association*, 81(394):461–470, 1986.

[11] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *Proceedings of the Second European Conference on Computational Learning Theory*, EuroCOLT '95, pages 23–37, London, UK, UK, 1995. Springer-Verlag.

[12] Jerome H. Friedman. Greedy Function Approximation: A Gradient Boosting Machine. In *Annals of Statistics*, volume 29, pages 1189–1232, 2000.

[13] K. Friston, L. Harrison, J. Daunizeau, S. Kiebel, C. Phillips, N. Trujillo-Barreto, R. Henson, G. Flandin, and J. Mattout. Multiple sparse priors for the m/eeg inverse problem. *Neuroimage*, 39(3):1104–20, 2008.

[14] G. Golub et al. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

[15] I. F. Gorodnitsky, J. S. George, and B. D. Rao. Neuromagnetic source imaging with focuss: a recursive weighted minimum norm algorithm. *Electroencephalogr Clin Neurophysiol*, 95(4):231–51, 1995.

[16] A. Gramfort, M. Kowalski, and M. Hamalainen. Mixed-norm estimates for the m/eeg inverse problem using accelerated gradient methods. *Phys Med Biol*, 57(7):1937–61, 2012.

[17] R. Grech, T. Cassar, J. Muscat, K. P. Camilleri, S. G. Fabri, M. Zervakis, P. Xanthopoulos, V. Sakkalis, and B. Vanrumste. Review on solving the inverse problem in eeg source analysis. *J Neuroeng Rehabil*, 5:25, 2008.

[18] M. Hamalainen, R. Ilmoniemi, J. Knuutila, and O. Lounasmaa. Magnetoencephalography: theory, instrumetation and applications to the non-invasive study of human brain function. *Rev. Mod. Phys.*, 65:413–497, 1993.

[19] R. N. Henson, D. G. Wakeman, V. Litvak, and K. J. Friston. A parametric empirical bayesian framework for the eeg/meg inverse problem: Generative models for multi-subject and multi-modal integration. *Front Hum Neurosci*, 5:76, 2011.

[20] AC Huk, RF Dougherty, and DJ Heeger. Retinotopy and functional subdivision of human areas mt and mst. *J Neurosci*, 22:7195–7205, 2002.

[21] 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*, ICML '09, pages 433–440, New York, NY, USA, 2009. ACM.

[22] K. Kato. On the degrees of freedom in shrinkage estimation. *Journal of Multivariate Analysis*, 100 (7):1338–1352, 2009.

[23] Yehuda Koren. Collaborative filtering with temporal dynamics. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD'09)*, 2009.

[24] V. Litvak and K. Friston. Electromagnetic source reconstruction for group studies. *Neuroimage*, 42(4):1490–8, 2008.

[25] E. Martinez-Montes et al. Identifying complex brain networks using penalized regression methods. *J. Biol. Phys*, 34(3-4):315–323, 2008.

[26] K. Matsuura and Y. Okabe. Selective minimum-norm solution of the biomagnetic inverse problem. *IEEE Trans Biomed Eng*, 42(6):608–15, 1995.

[27] J. Mattout, R. N. Henson, and K. J. Friston. Canonical source reconstruction for meg. *Comput Intell Neurosci*, page 67613, 2007.

[28] MB Miller, M Li, G Lind, and S-Y Jang. Problem 3: Simulated rheumatoid arthritis data. *BMC Proceedings 1 (Suppl1):S4*, 2007.

[29] B. O'Donoghue and E. Candes. Adaptive restart for accelerated gradient schemes. *Foundations of Computational Mathematics*, 2012.

[30] W. Ou, M. S. Hamalainen, and P. Golland. A distributed spatio-temporal eeg/meg inverse solver. *Neuroimage*, 44(3):932–46, 2009.

[31] R. D. Pascual-Marqui, C. M. Michel, and D. Lehmann. Low resolution electromagnetic tomography: a new method for localizing electrical activity in the brain. *Int J Psychophysiol*, 18(1):49–65, 1994.

[32] C. Phillips, J. Mattout, M. D. Rugg, P. Maquet, and K. J. Friston. An empirical bayesian solution to the source reconstruction problem in eeg. *Neuroimage*, 24(4):997–1011, 2005.

[33] LeBlanc ML Ruczinski I, Kooperberg C. Logic regression. *Journal of Computational and Graphical Statistics*, 12(3):475–511, 2003.

[34] S. Rush and D. A. Driscoll. Eeg electrode sensitivity–an application of reciprocity. *IEEE Trans Biomed Eng*, 16(1):15–22, 1969.

[35] L. Scheffe. *The Analysis of Variance.* Wiley, 1959.

[36] Holger Schwender and Katja Ickstadt. Identification of snp interactions using logic regression. *Biostatistics*, 9(1):187–198, 2008.

[37] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani. A sparse-group lasso. *Journal of Computational and Graphical Statistics*, 22(2):231–245, 2013.

[38] SM Smith. Fast robust automated brain extraction. *Hum Brain Mapp*, 17:143–155, 2002.

[39] A Tarantola. *Inverse Problem Theory and Methods for Model Parameter Estimation*. Society for Industrial and Applied Mathematics, Philadelphia, 2005.

[40] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society - Series B: Statistical Methodology*, 58(1):267–288, 1996.

[41] R. Tibshirani et al. Strong rules for discarding predictors in lasso-type problems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2012.

[42] RB Tootell and N Hadjikhani. Where is dorsal v4 in human visual cortex? retinotopic, topographic and functional evidence. *Cereb Cortex*, 11:298–311, 2001.

[43] K. Uutela, M. Hamalainen, and E. Somersalo. Visualization of magnetoencephalographic data using minimum current estimates. *Neuroimage*, 10(2):173–80, 1999.

[44] Jr. Vaughan, H. G. and W. Ritter. The sources of auditory evoked responses recorded from the human scalp. *Electroencephalogr Clin Neurophysiol*, 28(4):360–7, 1970.

[45] AR Wade, AA Brewer, JW Rieger, and BA Wandell. Functional measurements of human ventral occipital cortex: Retinotopy and colour. *Philos Trans R Soc Lond B Biol Sci*, 357:963–973, 2002.

[46] J. Z. Wang, S. J. Williamson, and L. Kaufman. Magnetic source images determined by a lead-field analysis: the unique minimum-norm least-squares estimation. *IEEE Trans Biomed Eng*, 39(7):665–75, 1992.

[47] D. Wipf and S. Nagarajan. A unified bayesian framework for meg/eeg source imaging. *Neuroimage*, 44(3):947–66, 2009.

[48] D. P. Wipf, J. P. Owen, H. T. Attias, K. Sekihara, and S. S. Nagarajan. Robust bayesian estimation of the location, orientation, and time course of multiple correlated neural sources using meg. *Neuroimage*, 49(1):641–55, 2010.

[49] 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.

[50] P. Zhao, G. Rocha, and B. Yu. Grouped and hierarchical model selection through composite absolute penalties. *The Annals of Statistics*, 37 (6A):3468–3497, 2009.

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

Michael Lim

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Trevor Hastie) Principal Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Robert Tibshirani)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Jonathan Taylor)

Approved for the University Committee on Graduate Studies