Theory and Methods for High Dimensional Structured Pattern Recovery

By

Nikhil Surendra Rao

A dissertation submitted in partial fulfillment of

the requirements for the degree of

Doctor of Philosophy

(Electrical and Computer Engineering)

at the

UNIVERSITY OF WISCONSIN – MADISON

2014

Date of final oral examination: 6/26/2014

The dissertation is approved by the following members of the Final Oral Committee:
Robert Nowak, Professor, Electrical and Computer Engineering
Stephen Wright, Professor, Computer Science
Barry Van Veen, Professor, Electrical and Computer Engineering
Timothy Rogers, Professor, Psychology
Rebecca Willett, Associate Professor, Electrical and Computer Engineering

If you torture data enough, it will confess

–Ronald Coase

Abstract

In the past few years, there has been an exponential growth in the amount of data collected in many fields. Performing statistical inference on data on a large scale brings with it many challenges. To deal with these challenges, data is typically represented using components that are “simple". These notions of simplicity typically allow tractable methods to be used to perform inference on the data. This thesis focuses on understanding algorithmic, statistical and theoretical questions that arise in large-scale structured model selection problems using (big) data.

This thesis can be broadly categorized into three parts. The first portion of this thesis involves theoretical contributions for structured sparse signal recovery. The lasso has been a widely used tool to recover signals that are sparse, and the group lasso is a natural extension for signals that exhibit structure amongst the non zero components. However, the group lasso cannot handle the case of overlapping groups efficiently and hence finds limited use. We analyze a method that was proposed to overcome the aforementioned drawback of the group lasso. We derive sample complexity bounds for the group lasso with overlap (also called the latent group lasso), and show that the number of measurements needed only depends on the size and number of groups, and not the complexity of overlap between the groups. Furthermore, motivated by applications in functional Magnetic Resonance Imaging and computational biology, we introduce the Sparse Overlapping Sets lasso (SOSlasso) that can recover signals that are not only (overlapping) group sparse, but many components within a group are also zero. We derive sample complexity bounds for the SOSlasso in linear and logistic regression settings. The SOSlasso generalizes the group lasso with overlap, and can be used to recover structured

patterns spanning a wide range of applications.

We then turn to algorithms for recovering signals that are simple in a very general sense of the word. The algorithmic framework that we propose can be used for standard sparse recovery, group sparse recovery, low rank matrix completion methods, group sparse regularized problems in multitask learning, among others. The algorithm can also be used to recover signals in cases where no tractable methods exist, such as super resolution applications in signal processing, or cases where existing methods are intractable due to massive memory requirements, such as the group lasso with overlapping groups. The method can also be used to perform regression on large graphs, where the graph can be decomposed into (overlapping) edges, cycles and/or cliques. Also, one can use our method to recover signals that are made up of a combination of different structures.

Lastly, this thesis focuses on novel applications that involve structured pattern recovery. We show the utility of the SOSlasso on multitask learning in fMRI and gene selection applications in computational biology. We then show a novel modeling scheme for recovering wavelet transform coefficients in inverse problems. Our method to model the coefficients allows us to solve convex recovery problems, while at the same time taking advantage of the structure inherent in the coefficients.

We finally conclude the thesis and discuss some ongoing research, and also discuss extensions to the work presented.

Acknowledgments

Nearly six years ago, I decided to pursue graduate studies in the United States immediately after obtaining my bachelor's degree. Little did I know what would be in store for me, and I definitely did not expect what it actually ended up being. Research was definitely challenging, but at the same time extremely gratifying, and my time at UW Madison would not be as memorable if not for the constant support of friends and family.

To my fellow lab mates Gautam Dasarathy, Aniruddha Bhargava, Matt Malloy, Kevin Jamieson, Yana Shkel and Zac Harmany, a big thank you. We have had endless conversations on innumerable topics, in the lab and in bars, and of course at the Terrace. My time here would not be nearly as rewarding had it not been for the wonderful moments I have shared with you.

Parikshit Shah, Chris Cox and Tim Rogers, thank you for the wonderful discussions we have had over various research topics. Interacting with you has allowed me to think about and understand problems in domains beyond my supposed expertise, and the experience has been enriching.

I am forever indebted to all the professors that have helped me along the way. To Steve Wright, thank you for many helpful discussions, and for being not only a collaborator and painstakingly going over code and proofs, but also providing sound advice about various topics in optimization.

To my advisor Rob Nowak: words cannot express how much I have learnt from you over the years. Thank you for creating and nurturing an environment that allows students to freely mingle with each other, and professors, and share ideas, for teaching me to be extremely rigorous about my work, and to not assume that the answer is known until I have found it out

for myself.

And finally, to my family. Without your constant support and unconditional love, none of this would be possible. To my parents, thank you for everything you have given me over the years. For the support you have given me throughout childhood right up to this moment, I have accrued a debt that can never be repaid. Thank you for ensuring that I do not get caught up with work, and teaching me to keep life in perspective. Varun, thank you for being the best brother a guy can ask for. And to my fiancee Parvi for the wonderful words of encouragement. Countless times you have kept faith in me and motivated me when I have felt disappointed with results. You are the most amazing person in the world, and I cannot express how much I love you.

Contents

Chapter 1

Introduction

In recent years, there has been an explosion of data collected in diverse fields such as genetics, business, social media, physics and engineering. Almost all modern signal processing and machine learning applications have to deal with massive data, and this leads to daunting challenges for statistical inference. However, in many cases, salient aspects of the data can be represented in terms of a small number of “simple" components. These components might correspond to blocks of values arising from correlations between transform domain coefficients in signal processing, spatial or functional connectivity of voxels in cross-subject fMRI studies, components connected by edges in a graph, low-rank structure in designing recommendation systems, across task feature similarities in multitask learning applications, genes belonging to a particular pathway, among other applications. This thesis focuses on understanding algorithmic, statistical and theoretical questions that arise in large-scale structured model selection problems.

Dealing with data on a large scale brings with it two major challenges. In applications such as neuroscience, genetics and signal processing, the process of acquiring data itself is often a costly affair. The “cost" here might refer to the time taken to obtain measurements, monetary cost constraints, available power constraints or a combination of these. Solving the inference/recovery problem with small amounts of measurements becomes very hard, or downright impossible. To alleviate this, one often needs to model the data using much simpler

components, so that the limited measurements on hand suffice to recover the signal of interest. Such modeling assumptions typically manifest themselves in the form of sparsity and group sparsity in signal processing and low rank assumptions in matrix completion and recommendation systems. These assumptions on the model make intractable problems tractable, and facilitate the use of efficient algorithms to perform inference. While sparsity is the most common assumption made to enforce structure on the data, in some cases one has additional information about the kinds of sparsity patterns present. In this thesis, we focus on structure in the sparsity pattern. Motivated by problems in functional Magnetic Resonance Imaging, computational biology and image processing, we focus on different notions of overlapping group sparse models. We show that when additional information about the sparsity pattern is known, the number of measurements needed to solve high dimensional structurally constrained problems is less than that needed without these assumptions, while at the same time allowing for efficient convex recovery methods to be used. We apply the methods we develop to the aforementioned applications, and show that we indeed perform better than methods where such assumptions are not taken into account, either in the modeling of the data or in the reconstruction algorithm used.

The second challenge that arises in most modern-day machine learning applications is that traditional optimization techniques might not be well suited to handle arbitrary notions of simplicity in the data. While efficient algorithms exist for recovering signals with certain predefined structure (for example sparsity and low rank), a method that works for fairly arbitrary notions of “simplicity" has not been forthcoming. For example, one might wish to represent the data as being made up of a small number of edges on a graph. Most first order methods that exist to solve this problem involve huge computational costs. We look to bridge this gap in this thesis. We develop a method that retains the computational efficiency of greedy methods,

while at the same time delivering state of the art performance on many different kinds of signals that have a simple structure with respect to a predefined (but arbitrary) basis or frame. We show that the method enjoys nice theoretical convergence properties, and apply the algorithm to a vast variety of problems in signal processing and machine learning, proving its utility in many real world applications.

1.1 Summary of Contributions

We summarize our contributions in this section. We give a brief overview of the Chapters in the sequel, and also point the interested reader to relevant publications that (s)he may peruse.

Chapter 2

In this chapter, we derive theoretical results for structured sparse signal recovery. Standard compressive sensing results state that to exactly recover an s sparse signal in $\mathbb{R}^p$, one requires $\mathcal{O}(s \cdot \log p)$ measurements. While this bound is extremely useful in practice, often real world signals are not only sparse, but also exhibit structure in the sparsity pattern. We focus on group structured sparsity patterns first. Under this model, groups of signal coefficients are active (or inactive) together. The groups are predefined, but the particular set of groups that are active (i.e., in the signal support) must be learned from measurements. We show that exploiting knowledge of groups can further reduce the number of measurements required for exact signal recovery, and derive universal bounds for the number of measurements needed. The bound is universal in the sense that it only depends on the number of groups under consideration, and not whether the groups overlap. Experiments show that our result holds for a variety of overlapping group configurations.

In many applications, however, the group lasso with overlapping groups itself can be very restrictive. We hence turn our attention to a less restrictive form of structured sparse feature selection: we assume that while features can be grouped according to some notion of similarity, not all features in a group need be selected for the task at hand. We introduce a new procedure called *Sparse Overlapping Sets (SOS) lasso*, a convex optimization program that automatically selects similar features for learning in high dimensions. We establish consistency results for the SOSlasso for classification problems using the logistic regression setting, which specializes to results for the lasso and the group lasso, some known and some new. We also prove sample complexity bounds in linear regression settings. In particular, SOSlasso is motivated by multi-subject fMRI studies in which functional activity is classified using brain voxels as features, source localization problems in Magnetoencephalography (MEG), and analyzing gene activation patterns in microarray data analysis.

Relevant Publications:

- N. Rao, R. Nowak, C. Cox and T. Rogers *Logistic Regression with Structured Sparsity*, arXiv:1402.4512 , 2014 (In preparation for submission to the Journal of Machine Learning Research)
- N. Rao, C. Cox, R. Nowak, and T. Rogers *Sparse Overlapping Sets Lasso for Multitask Learning and fMRI Data Analysis*, NIPS, 2013
- N. Rao, B. Recht and R. Nowak *Universal Measurement Bounds for Structured Sparse Signal Recovery*, AISTATS, 2012

Chapter 3

In many signal processing and machine learning applications, one aims to reconstruct a signal that has a simple representation with respect to a certain basis or frame. Fundamental elements of the basis known as "atoms" allow us to define "atomic norms" (to be explained in the sequel shortly) that can be used to construct convex regularizers for the reconstruction problem. Efficient algorithms are available to solve the reconstruction problems in certain special cases, but an approach that works well for general atomic norms remains to be found. This chapter describes an optimization algorithm called CoGEnT , which produces solutions with succinct atomic representations for reconstruction problems, generally formulated with atomic-norm constraints. CoGEnT combines a greedy selection scheme based on the conditional gradient approach with a backward (or "truncation") step that exploits the quadratic nature of the objective to reduce the basis size. We establish convergence properties and validate the algorithm via extensive numerical experiments on a suite of signal processing applications. Our algorithm and analysis are also novel in that they allow for *inexact* forward steps. In practice, CoGEnT significantly outperforms the basic conditional gradient method, and indeed many methods that are tailored to specific applications. We also introduce several novel applications that are enabled by the atomic-norm framework, including tensor completion and moment problems in signal processing.

Relevant Publications:

- N. Rao, P. Shah and S. Wright *Forward-backward Greedy Algorithms for Atomic Norm Regularization*, arXiv:1404.5692, 2014 (submitted to IEEE Trans. Signal Processing)

- N. Rao, P. Shah and S. Wright *Forward-backward Greedy Algorithms for Signal Demixing*, ASILOMAR (Invited Paper), 2014

- N. Rao, P. Shah, S. Wright and R. Nowak *A Greedy Forward-backward Algorithm for Atomic Norm Constrained Minimization*, ICASSP, 2013

Chapter 4

In this chapter, we exclusively focus on applications that motivated the theoretical work of the first chapter. A major motivating application for the Sparse Overlapping Sets lasso is the analysis of multi-subject fMRI data. We show that the SOSlasso is ideally suited to simultaneously take advantage of large scale inter subject similarities and the small scale dissimilarities between subjects. Experimental results validate our claim, and we achieve better classification accuracy and more meaningful results when compared to the lasso, elastic net and different versions of the group lasso. Along similar lines, we postulate that the SOSlasso can be used to improve performance on gene microarray datasets, and validate our claim experimentally.

We then turn our attention to modeling wavelet coefficients in images. Statistical dependencies among wavelet coefficients are commonly represented by graphical models such as hidden Markov trees (HMTs). However, in linear inverse problems such as deconvolution, tomography, and compressed sensing, the presence of a sensing or observation matrix produces a linear mixing of the simple Markovian dependency structure. This leads to reconstruction problems that are non-convex optimizations. Past work has dealt with this issue by resorting to greedy or suboptimal iterative reconstruction methods. In this chapter, we propose new modeling approaches based on group-sparsity penalties that leads to convex optimizations that can be solved exactly and efficiently. We show that the methods we develop perform significantly

better in deconvolution and compressed sensing applications, while being almost as computationally efficient as standard coefficient-wise approaches such as lasso. As an extension to the method we propose, we also investigate a scheme to design the sensing matrix in compressed sensing applications, when additional information about the sparsity pattern is known a priori. We show that by designing a sensing matrix that is “well aligned" with the subspace in which the signal is expected to lie in, we get a performance boost due to increased Signal to Noise Ratios.

Relevant Publications:

- N. Rao, C. Cox, R. Nowak, and T. Rogers *Sparse Overlapping Sets Lasso for Multitask Learning and fMRI Data Analysis*, NIPS, 2013
- N. Rao, R. Nowak, S. Wright and N. Kingsbury *Convex Approaches to Model Wavelet Sparsity Patterns* ICIP 2011
- N. Rao and R. Nowak *Correlated Gaussian Designs for Compressive Imaging*. ICIP 2012

1.1.1 Full List of Publications

1. *Logistic Regression with Structured Sparsity*: Nikhil Rao, Robert Nowak, Chris Cox and Timothy Rogers, arXiv:1402.4512 , 2014 (In preparation for submission to the Journal of Machine Learning Research)
2. *A Forward-Backward Algorithm for Atomic Norm Regularization*: Nikhil Rao, Parikshit Shah and Stephen Wright, arXiv:1404.5692, 2014 (submitted to IEEE Trans. Signal Processing)

3. *Forward-Backward Greedy Algorithms for Signal Demixing*: Nikhil Rao, Parikshit Shah and Stephen Wright, Asilomar Conference on Signals, Systems and Computers (Invited paper), 2014

4. *Sparse Overlapping Sets Lasso for Multitask Learning and fMRI Data Analysis*: Nikhil Rao, Christopher Cox, Robert Nowak and Timothy Rogers, Neural Information Processing Systems 2013 (Spotlight Presentation)

5. *Conditional Gradient with Enhancement and Truncation for Atomic Norm Regularization*: Nikhil Rao, Parikshit Shah and Stephen Wright, NIPS workshop on Greedy Algorithms 2013

6. *A Greedy Forward Backward Method for Atomic Norm Constrained Minimization*: Nikhil Rao, Parikshit Shah, Stephen Wright and Robert Nowak, IEEE International Conference on Acoustics, Speech and Signal Processing, 2013

7. *Adaptive Sensing with Structured Sparsity*: Nikhil Rao, Gongguo Tang and Robert Nowak, IEEE International Conference on Acoustics, Speech and Signal Processing, 2013

8. *Knowledge Enhanced Measurements for Estimating Sparse Signals from Clutter*: Workshop on Signal Processing with Adaptive Structured Sparse Representations (SPARS) 2013, Lausanne, Switzerland

9. *Adaptive Sensing on Markov Trees* : Workshop on Signal Processing with Adaptive Structured Sparse Representations (SPARS) 2013, Lausanne, Switzerland

10. *Correlated Gaussian Designs for Compressive Imaging* : Nikhil Rao and Robert Nowak, IEEE International Conference on Image Processing, 2012

11. *Universal Measurement Bounds for Structured Sparse Signal Recovery* : Nikhil Rao, Benjamin Recht and Robert Nowak, Journal of Machine Learning Research (proc. Artificial Intelligence and Statistics), 2012

12. *A Clustering Approach to Optimize Online Dictionary Learning* : Nikhil Rao and Fatih Porikli, IEEE International Conference on Acoustics, Speech and Signal Processing, 2012

13. *Convex Approaches for Group Sparse Signal Recovery in Compressed Sensing* : Workshop on Signal Processing with Adaptive Structured Sparse Representations (SPARS) 2011, Edinburgh, UK

14. *Convex approaches to Model Wavelet Sparsity Patterns* : Nikhil Rao, Robert Nowak, Stephen Wright and Nick Kingsbury, IEEE international Conference on Image Processing, 2011 (1st prize, Best Student Paper Award)

15. *Using Machines to Improve Human Saliency Detection* : Nikhil Rao, Joseph Harrison, Tyler Karrels, Robert Nowak and Timothy Rogers , Asilomar Conference on Signals, Systems and Computers, 2010

1.2 Background on Sparse and Structured Sparse Signal Recovery

We first set up some preliminary concepts that will make an appearance throughout this thesis. The notion of sparse pattern recovery has played a central role in many signal processing and machine learning applications. Here, a high dimensional target vector [1] is assumed to be sparse, meaning only a small fraction of the entries are non zero. As a running example, consider the problem of inferring what genes are responsible for causing metastasis in breast cancer tumors, from microarray data. The number of potential genes in a sample is typically in the thousands, whereas the number of samples available for the experiment will be at most in the hundreds. A natural assumption to make is that a biologist will only be interested in a small fraction of the thousands of genes. Indeed, typically the number of relevant genes is usually in the order of tens. The idea is then to recover the subset of genes that are relevant for diagnosis, from a small number of observations, typically in the hundreds.

The most natural optimization framework to recover a sparse vector is the following ℓ_0 pseudo norm penalized program:

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) \;\; \textbf{s.t.} \;\; \|\boldsymbol{x}\|_0 \leq s \tag{1.1}$$

Where $f(\cdot)$ is a (convex) loss function. Typical examples for $f(\cdot)$ include the least squares loss and the logistic loss.

The ℓ_0 penalty is non convex, but admits a convex relaxation in terms of the ℓ_1 norm. In fact, the ℓ_1 norm is a tight convex relaxation of the ℓ_0 pseudo norm, and the corresponding

[1] We assume that matrices and tensors are vectorized

convex optimization problem is given by

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) \;\; \textbf{s.t.} \; \|\boldsymbol{x}\|_1 \leq s \tag{1.2}$$

or equivalently the Lagragian form is given by,

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) + \lambda \|\boldsymbol{x}\|_1 \tag{1.3}$$

(1.2) and (1.3) have found use in a wide variety of situations, and have been extensively studied in the literature [13, 24, 58, 83]. [83] referred to the above convex optimization problem as the lasso (Least Absolute Shrinkage and Selection Operator). The lasso [2] is useful for a couple of reasons: In many cases, the solution is extremely high dimensional, and even for solving a simple system of linear equations, it becomes prohibitive to obtain many measurements. In such scenarios, the problem is heavily underdetermined. Fortunately, if it is known that the high dimensional solution is sparse, then there is hope, since the number of unknowns is far less than the problem dimension, and meaningful recovery is possible. In our running example, this will correspond to selecting a small number of genes to be relevant for prediction of metastasis. In other cases, even if we can solve the system of equations, the solution makes little sense if there are few (if any) zeros in it, since the idea is to recover a small subset of explanatory variables for the data at hand.

While the importance and utility of the lasso to a wide variety of applications in signal processing and machine learning is unquestioned, in many cases one has additional information about the problem. The simplest form of this information is the knowledge that certain

[2] We use the term lasso to mean both the penalized and the Lagrangian formulations, since they are equivalent.

features [3] are highly correlated with each other. These correlated features form blocks in the feature space, with the following effect: if, from a certain group, a single feature is relevant, then all the features from that group are relevant. In our running example, this means that it is known that certain genes are highly correlated with each other, and so it is highly likely that if a certain gene is relevant, then the genes that are correlated are also relevant.

Thus, the goal is to not only select a small number of features, as in the lasso, but to select a small number of entire groups of features (See Figure 1 for an illustration). The group lasso [91] was proposed to achieve this goal. The group lasso can be written as the following convex program

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) + \lambda \sum_{G \in \mathcal{G}} \|\boldsymbol{x}_G\|_2 \tag{1.4}$$

where $G \in \mathcal{G}$ is a known set of groups, so that $G_i \cap G_j = \{\}, \;\; i \neq j$, and $\boldsymbol{x}_G$ is the sub vector of $\boldsymbol{x}$ indexed by group G.

(a) Sparsity pattern promoted by the lasso. Many elements are set to zero

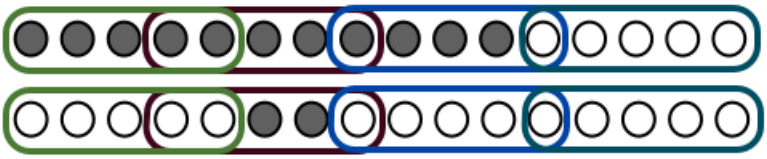
(b) Sparsity pattern when certain groups of variables are highly correlated. The large ovals indicate groups. Note that some groups are set entirely to zero

Figure 1: Sparsity patterns promoted by the lasso (a), and that needed when groups of features need to be jointly selected (b), and promoted by the group lasso

In many applications of interest however, the groups overlap with each other. Coming back to our running example in computational biology, it is reasonable to assume that not only are groups of genes correlated with each other, but there exist genes that are part of multiple sets

[3] We use the term variables and features interchangeably in this work. In a strict sense, variables correspond to the elements of the vector $\boldsymbol{x}$, while features correspond to the data observed. However, it will be clear from context whether we are referring to the elements in the solution itself, or the data observed.

(a) A vector with overlapping groups. The second group (indexed in red) is active



(b) Potential sparsity patterns obtained by solving the group lasso. Since entire groups are set to zero, we will either select one group or three. In either case, the resulting sparsity pattern is incorrect

Figure 2: A vector with overlapping groups, and sparsity patterns obtained from the group lasso. Note that in (b), some groups are entirely set to zero. The pattern of zeros form a union of groups, and hence the non zero pattern is a complement of a union of groups.

. This has in fact been observed in real datasets [80]. The group lasso (1.4) is inappropriate for use in such situations. The group lasso sets entire groups to zero, and hence the recovered sparsity pattern is a complement of a union of groups (Figure 2). However, a more natural way to think about the group lasso is that we want a union of groups to be nonzero, as in Figure 3. Coming back to our running example, what we want is to select sets of genes that are correlated with each other, regardless of whether one of those genes lies in another set that is not selected.

To select sparsity patterns that can be cast as a union of groups, [37] proposed the group lasso with overlap (also called the latent group lasso). It was shown that this method selects sparsity patterns that are unions of groups, and thus can be used for a wide variety of applications. In this thesis, we study this formulation further, and provide sample complexity bounds for the group lasso with overlap in Chapter 2. In Chapter 3, we also propose an algorithm that can be used to solve this optimization problem, with a much smaller memory footprint than traditional proximal point based methods.

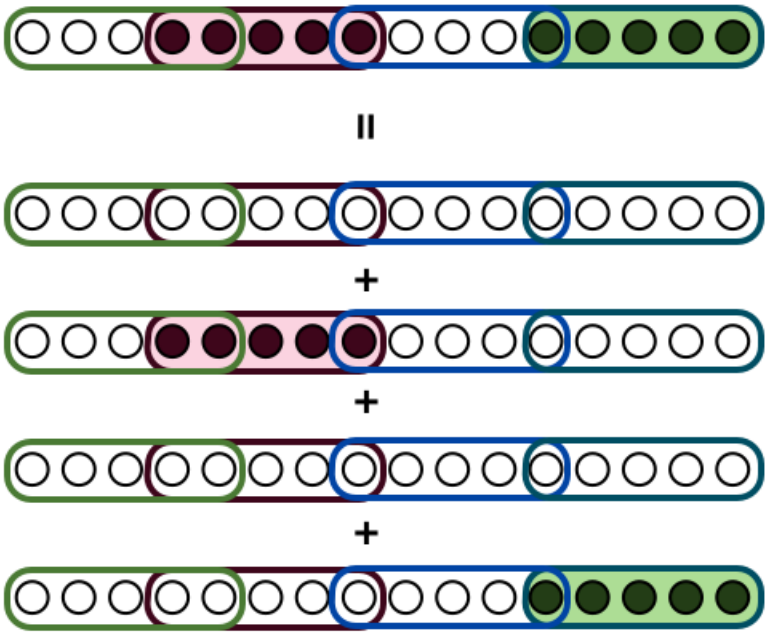


Figure 3: Decomposition of a vector into latent vectors. Each vector corresponds to one of the four groups present in the signal. The groups indexed in red and green are active, and the resulting sparsity pattern is realized by activating the corresponding vectors, and adding them up.

While the group lasso with overlapping groups allows us to recover sparsity patterns that can be seen to be a union of groups, there are still some applications where this might not be ideal. Again, going back to our running example in computational biology, not only are correlated genes arranged in overlapping groups, but each group contains 100's of genes. It is unreasonable to assume that if a certain group is active, then each and every one of the genes in the group are relevant. This motivates a method that not only selects a union of groups, but once a group is selected, only a small fraction of the variables within that group is activated, as illustrated in Figure 4.

Figure 4: Sparsity pattern where a single group is active (among overlapping groups) and within the active group, only a few coefficients are active.

In Chapter 2, we introduce a method that does precisely this. We call it the Sparse Overlapping Sets (SOS) lasso, and provide a theoretical analysis of the same, In Chapter 4, we provide some experimental results and show that the SOSlasso outperforms other standard methods for sparse regression in some applications of interest. We also show that the SOSlasso is a generalization of the lasso and the (overlapping) group lasso, and specific choices of parameters lead to the lasso or the group lasso formulations.

1.3 Solving the Overlapping Group Lasso Problem

In the previous section, we introduced the group lasso with overlapping groups, as an extension to the standard group lasso, where the groups form a partition of the ambient space. The

optimization problem for the group lasso with overlapping groups is given by

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} f(\boldsymbol{x}) + \lambda \Omega^{\mathcal{G}}_{overlap}(\boldsymbol{x}) \tag{1.5}$$

where, for a given set of groups $\mathcal{G}$, $\Omega^{\mathcal{G}}_{overlap}(\boldsymbol{x})$ is the overlapping group lasso penalty given by

$$\Omega^{\mathcal{G}}_{overlap}(\boldsymbol{x}) = \inf \sum_{G \in \mathcal{G}} \boldsymbol{w}_G \;\; \textbf{s.t.} \;\; \boldsymbol{x} = \sum_{G \in \mathcal{G}} \boldsymbol{w}_G \tag{1.6}$$

where $\boldsymbol{w}_G$ is a vector with the same length as $\boldsymbol{x}$, but with zeros in the locations not indexed by G.

In all the applications we are concerned about in this thesis, and indeed many practical applications, the ambient dimension is large ($\boldsymbol{x} \in \mathbb{R}^p$ with p large). In such scenarios, second order methods become prohibitive very quickly, and hence first order methods are a common choice. Among first order methods, the proximal gradient algorithm, and its accelerated versions is one of the most widely used. For a problem of the form

$$\min f(\boldsymbol{x}) + \lambda g(\boldsymbol{x})$$

where $f(\boldsymbol{x})$ is smooth and convex is $\boldsymbol{x}$, and $g(\boldsymbol{x})$ is potentially non smooth but convex in $\boldsymbol{x}$, the basic proximal point method is an iterative procedure that follows the following steps in each iteration (η being an appropriately chosen step size):

$$\boldsymbol{x}^{t+\frac{1}{2}} = \boldsymbol{x}^t - \eta \nabla f(\boldsymbol{x}^t)$$

$$\boldsymbol{x}^{t+1} = \text{prox}_{\lambda g}(\boldsymbol{x}^{t+\frac{1}{2}})$$

where $\text{prox}_g(\cdot)$ is the proximal point operator obtained as the solution to the following convex program

$$\text{prox}_g(\boldsymbol{x}) = \arg\min_{\boldsymbol{y}} \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{y}\|^2 + \lambda g(\boldsymbol{y})$$

When the function $g(\boldsymbol{x})$ is the ℓ_1 norm, the proximal point function is the standard soft thresholding operator:

$$g(\boldsymbol{x}) = \|\boldsymbol{x}\|_1 \Rightarrow \text{prox}_g(\boldsymbol{x}) = \text{sign}(\boldsymbol{x})[|\boldsymbol{x}| - \eta\lambda]^+$$

where $[\cdot]^+$ indicates an element wise thresholding operation of the form $\max(0, \cdot)$. For $g(\cdot)$ being the group lasso norm, we have on a per group basis

$$g(\boldsymbol{x}) = \sum_{G \in \mathcal{G}} \|\boldsymbol{x}_G\| \Rightarrow (\text{prox}_g(\boldsymbol{x}))_G = \frac{\boldsymbol{x}_G}{\|\boldsymbol{x}_G\|}\left[\|\boldsymbol{x}_G\| - \eta\lambda\right]^+ \tag{1.7}$$

The proximal point operator for the group lasso (1.7) gives further insights into the kinds of sparsity patterns promoted by the group lasso. We see that, (1.7) sets entire groups to zero, if the corresponding group norm is larger than a certain threshold ($\eta\lambda$). Hence, a number of groups are entirely set to zero, meaning a union of groups is zeroed out. This results in the sparsity pattern (the set of non zero variables) to be expressed as a complement of a union of groups. When the groups are non overlapping, the complement of a union of groups is a union of another set of groups, but this is not true when the groups overlap. The group lasso with overlapping groups was introduced to overcome this precise drawback.

For $g(\cdot) = \Omega^{\mathcal{G}}_{overlap}(\cdot)$, a simple closed form for the proximal operator does not exist. However, we can reduce the problem to the standard group lasso problem, by replicating the variables that are part of more than one group. Specifically, we form a new vector $\tilde{\boldsymbol{x}} \in$

$\mathbb{R}^{\sum_{G\in\mathcal{G}}|G|}$ as follows:

$$\tilde{\boldsymbol{x}} = [\boldsymbol{x}_{G1}^T \;\; \boldsymbol{x}_{G2}^T \ldots \boldsymbol{x}_{GM}^T]^T$$

The above vector lies in potentially a much larger space than $\mathbb{R}^p$, but by making copies of variables in multiple groups, we can now define a set of groups $\tilde{\mathcal{G}}$ that partitions $\mathbb{R}^{\sum_{G\in\mathcal{G}}|G|}$. We can then solve the standard group lasso problem in this lifted space:

$$\hat{\tilde{\boldsymbol{x}}} = \arg\min_{\tilde{\boldsymbol{x}}} f(\tilde{\boldsymbol{x}}) + \lambda \sum_{\tilde{G}\in\tilde{\mathcal{G}}} \|\tilde{\boldsymbol{x}}_{\tilde{G}}\|$$

and then recombine the variables to recover the optimal vector $\hat{\boldsymbol{x}}$. For more implementation details, we refer the interested reader to [37]. A second method, that does not involve explicit replication of variables is also possible [56] but we have observed that it does not yield any computational advantages.

1.4 Atoms and the Atomic Set

In the previous section, we saw how the group lasso with overlaps aims to overcome the drawbacks inherent in the lasso and the group lasso. In this section, we set up some preliminaries about a framework that unifies the methods mentioned above, and also a host of other applications in signal processing and machine learning.

Suppose $\boldsymbol{x}$ is the signal that we are interested in recovering, from noisy measurements. We assume that we can write

$$\boldsymbol{x} = \sum_i c_i \boldsymbol{a}_i, \;\; c_i \geq 0, \;\; \boldsymbol{a}_i \in \mathcal{A} \tag{1.8}$$

The set $\mathcal{A}$ will be called the atomic set, and its members will be called atoms. So, for

any signal of interest, we assume that it can be represented as a conic combination of atoms. Furthermore, we will mainly be interested in signals that can be expressed not only as in (1.8) but with the additional assumption that most of the $c_i's$ are zero:

$$\boldsymbol{x} = \sum_i c_i \boldsymbol{a}_i, \;\; c_i \geq 0, \;\; \boldsymbol{a}_i \in \mathcal{A}, \;\; \|\boldsymbol{c}\|_0 \leq s \tag{1.9}$$

where $\boldsymbol{c}$ is the vector whose elements are the $c_i's$.

The representation in terms of atoms allows us to work with a wide variety of signals:

1. When the atoms are signed canonical basis vectors, the signal can be seen to be sparse.
2. When the atoms are unit norm matrices, the signal will be a low rank matrix
3. When the atoms are unit vectors whose support is restricted to certain predefined indices, we obtain group sparse vectors.

Later in Chapter 3, we will give more examples of atoms, and the corresponding signals they give rise to. Armed with the atomic notation, the goal of recovering a signal that has a parsimonious representation with respect to a certain atomic set can be expressed as the solution to an optimization problem:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} f(\boldsymbol{x}) \quad \textbf{s.t.} \quad \|\boldsymbol{x}\|_{\mathcal{A},0} \leq s \tag{1.10}$$

where $\|\boldsymbol{x}\|_{\mathcal{A},0} := \|\boldsymbol{c}\|_0$, and $\boldsymbol{c}$ is per the representation in (1.9). We see that (1.10) is the equivalent of the ℓ_0 pseudo norm constrained problem for standard sparse vectors (1.1), but in a very general sense. Indeed, given an appropriately defined set of atoms $\mathcal{A}$, (1.10) yields a framework to recover signals that have a parsimonious representation in that set of atoms.

Specifically, from the representation (1.9), we have that $\boldsymbol{x}$ is a conic combination of atoms, and when $\|\boldsymbol{c}\|_0$ is small, it means that only a few atoms $\boldsymbol{a} \in \mathcal{A}$ are used to represent $\boldsymbol{x}$.

The above formulation is non convex however, and hence the problem is hard to solve in general. This motivates a convex relaxation of the above penalty, called the atomic norm

$$\|\boldsymbol{x}\|_{\mathcal{A}} = \inf \sum_i c_i \quad \textbf{s.t.} \quad \boldsymbol{x} = \sum_i c_i \boldsymbol{a}_i, \;\; c_i \geq 0, \;\; \boldsymbol{a}_i \in \mathcal{A} \tag{1.11}$$

The atomic norm is the gauge functional of the convex hull of the atoms in the set $\mathcal{A}$. When the set of atoms is centrally symmetric, $\|\boldsymbol{x}\|_{\mathcal{A}}$ is indeed a norm, and hence the name. For all the applications of interest in this thesis, we will have that $\mathcal{A}$ is symmetric about the origin, and hence we will continue to use the term “atomic norm".

The atomic norm essentially acts as the ℓ_1 equivalent of the function $\|\boldsymbol{x}\|_{\mathcal{A},0}$. The $\inf$ in the formulation ensures that when there are multiple sets of vectors $\boldsymbol{c}$ that satisfy $\boldsymbol{x} = \sum_i c_i \boldsymbol{a}_i$, we pick the one that has the smallest ℓ_1 norm. Note that the minimizer may not be unique, but it always exists. Returning to the examples listed above, we see that

1. When the atoms are canonical basis vectors, we obtain the ℓ_1 norm of the vector.

2. When the atoms are unit norm matrices, we obtain the nuclear norm of the matrix.

3. When the atoms are unit vectors whose support is restricted to certain predefined indices, we obtain the group lasso with overlap norm. The standard group lasso arises as a special case when the groups do not overlap.

We can then look to solve the following convex optimization problem

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} f(\boldsymbol{x}) \quad \textbf{s.t.} \quad \|\boldsymbol{x}\|_{\mathcal{A}} \leq s \tag{1.12}$$

The notion of atomic norms allows us to formulate recovery problems for a vast range of applications, and for fairly arbitrarily defined atomic sets $\mathcal{A}$. That being said, a practical algorithm that actually solves (1.12) has not been forthcoming. Indeed, for special cases of $\mathcal{A}$, very fast methods exist, but they cannot be easily generalized to arbitrary $\mathcal{A}$. In Chapter 3, we introduce an algorithm that solves this problem. The method enjoys nice theoretical convergence properties, is computationally efficient and can be applied to vast range of problems of practical interest in signal processing and machine learning.

Chapter 2

Sample Complexity Bounds for Learning with Structure

In this chapter, we theoretically analyze the problem of structured sparse signal recovery. First, we study the problem of regression with overlapping groups. We show that the group lasso with overlap can be cast as an atomic norm minimization problem (See Chapter 1 for an introduction to the atomic norm), and derive sample complexity bounds for the same. The bounds we derive yield a rather surprising result: the number of measurements needed for accurate recovery in an overlapping group sparse setting does not depend on the amount of overlaps between groups. It only depends on the number of groups and the group size, and of course the number of active groups.

We then motivate and introduce the Sparse Overlapping Sets (SOS) lasso. The SOSlasso generalizes the group lasso with overlapping groups and the lasso, and allows one to perform structured sparse recovery in a very flexible framework. We derive sample complexity bounds for the SOSlasso under both regression and classification settings, and show that under specific choices of parameters, we recover existing results for the lasso and the results for the overlapping group lasso.

The analysis for both the methods mentioned above relies on the notion of the mean (Gaussian) width, a robust geometrical property of sets. The problem of deriving sample complexity

bounds reduces to deriving bounds on the Gaussian width. We also validate our results on some toy data, leaving experiments on real datasets to Chapter 4.

2.1 Organization

We first derive sample complexity bounds for the group lasso with overlapping groups. We show that the problem can be cast as an atomic norm minimization algorithm, and show that by bounding the Gaussian width (to be defined) of a particular set, we can bound the number of measurements needed for exact recovery under noiseless settings, or approximate recovery in the presence of noise. The task of bounding the Gaussian width is non trivial, since there is no closed form expression for the set we will be interested in.

We then turn our attention to the problem of selecting sets of “similar" features, and not entire sets of features. We motivate applications where this might be necessary, and introduce the Sparse Overlapping Sets (SOS) lasso to solve inference problems with the aforementioned constraint. We derive consistency results for the SOSlasso under both regression and classification settings. We show that the SOSlasso generalizes the lasso and the group lasso, and thus provides a framework for structured signal recovery in a vast variety of applications.

2.2 Sample Complexity Bounds for the Group Lasso with Overlapping Groups

Standard compressive sensing results state that to exactly recover an s sparse signal in $\mathbb{R}^p$, one requires $\mathcal{O}(s \cdot \log p)$ measurements. While this bound is extremely useful in practice, often real world signals are not only sparse, but also exhibit structure in the sparsity pattern. We focus on

group-structured patterns in this section. Under this model, groups of signal coefficients are active (or inactive) together. The groups are predefined, but the particular set of groups that are active (i.e., in the signal support) must be learned from measurements. We show that exploiting knowledge of groups can further reduce the number of measurements required for exact signal recovery, and derive universal bounds for the number of measurements needed. The bound is universal in the sense that it only depends on the number of groups under consideration, and not the particulars of the groups (e.g., compositions, extents, overlaps, etc.). Experiments on toy data show that our result holds for a variety of overlapping group configurations.

2.2.1 Introduction

In many fields such as genetics, image processing, and machine learning, one is faced with the task of recovering very high dimensional signals from relatively few measurements. In general this is not possible, but fortunately many real world signals are, or can be transformed to be, sparse, meaning that only a small fraction signal coefficients are non-zero. Compressed Sensing [13, 24] allows us to recover sparse, high dimensional signals with very few measurements. In fact, results indicate that one only needs $\mathcal{O}(s \cdot \log p)$ random measurements to exactly recover an s sparse signal of length p.

In many applications however, one not only has knowledge about the sparsity of the signal, but some additional information about the structure of the sparsity pattern as well:

- In genetics, the genes are arranged into pathways, and genes belonging to the same pathway are often active/inactive in a group [80].

- In image processing, the wavelet transform coefficients can be modeled as belonging to a tree, with parent-child coefficients simultaneously being large or small [19, 73].

- In wideband spectrum sensing applications, the spectrum typically displays clusters of non-zero frequency coefficients, each corresponding to a narrowband transmission [55]

In cases such as these, the sparsity pattern can be represented as a union of certain groups of coefficients (e.g., coefficients in certain pathways, tree branches, or clusters). This knowledge about the signal structure can help further reduce the number of measurements one needs to exactly recover the signal. Indeed, the authors in [36] derive information theoretic bounds for the number of measurements needed for a variety of signal ensembles, including trees. In [5, 25], the authors show that one needs far fewer measurements when the signal can be expressed as lying in a union of subspaces, and explicit bounds are derived when using a modified version of CoSaMP [58] to recover the signal. In this chapter, we derive bounds on the number of random i.i.d. Gaussian measurements needed to exactly recover a sparse signal when its pattern of sparsity lies in a union of groups, when solving the *convex* recovery algorithm introduced in [37].

We analyze the group-structured sparse recovery problem using a random Gaussian measurement model. We emphasize that although the derivation assumes the measurement matrix to be Gaussian, it can be extended to *any* subGaussian case, by paying a small constant penalty, as shown in [**?**]. We restrict ourselves to the Gaussian case here since it highlights the main ideas and keeps the analysis as simple as possible.

Note that in this work, variables can be grouped into arbitrary sets, and we make *no* assumptions about the nature of the groups, except that they are known in advance. In short, we derive bounds for any generic group structure of variables, whether the groups overlap or form a partition of the ambient high dimensional space.

To the best of our knowledge, these results are new and distinct from prior theoretical characterizations of group lasso methods. Asymptotic consistency results are derived for the

group lasso when the groups partition the space of variables in [2]. Similarly, in [35], the authors consider the groups to partition the space, and derive conditions for recovery using the group lasso [91]. In [41, 42], the authors derive consistency results for the group lasso under arbitrary groupings of variables. In [59], the authors consider overlapping groups and derive sample bounds under the group lasso [91] setting. The authors in [37] derive consistency results in an asymptotic setting, for the group lasso with overlap, but do not provide exact recovery results. The general group lasso scenarios is different from what we consider, in that the group lasso yields vectors whose support can be expressed as a complement of a union of groups, while we consider cases where we require the support to lie in a union of groups, a distinction made in [37] and in Chapter 1 here. Note that in the case of non-overlapping groups, the complement of a union of groups is a union of (a different set of) groups. In this section, we (a) derive sample complexity bounds in a compressive-sensing framework when the measurement matrix is i.i.d. Gaussian. (b) We focus on non-asymptotic sample bounds, and in a case where the support is contained in a union of groups, and (c) make no assumptions about the nature of groups. To derive our results, we appeal to the notion of restricted minimum singular values of an operator.

We bound number of measurements needed for exact recovery with two terms. One term (kB) grows linearly in the total number of non-zero coefficients (with a small constant of proportionality). This is close to the bare minimum of one measurement per non-zero component. The other term only depends on the number of groups under consideration, and not the particulars of the groups (e.g., compositions, sizes, extents, etc.). In particular, the groups need not be disjoint. The degree to which groups overlap, remarkably, has no effect on our bounds. In this regard, our bounds can be termed to be *universal*. This is somewhat surprising since overlapping groups are strongly coupled in the observations, tempting one to suppose

that overlap may make recovery more challenging.

Our main result shows that for signals with support on k of M possible groups, exact recovery is possible from $(\sqrt{2\log(M-k)}+\sqrt{B})^2k+kB$ measurements using an overlapping group lasso algorithm, B being the maximum group size. Note that the bound depends on the sparsity s of the signal via the kB term. We will routinely compare the performance of the group lasso to the standard lasso, to study the effects of overlap between groups on the actual number of measurements needed to exactly recover a signal. For the lasso bound, we will use the one derived in [14]: $(2s+1)\log(p-s)$. Assuming that $M = \mathcal{O}(poly(p))$, our bound is roughly $k\log(p)+kB$. For the same problems, the lasso which ignores the group structure of the sparse signal components would require approximately $kB\log(p)$ measurements. Hence, taking advantage of the group structure will allow us to take fewer measurements to reconstruct the signal.

Our proof derives from the techniques developed in [14]. The rest of this section is organized as follows: in Section 2.2.2, we lay the groundwork for the main contribution of the chapter, *viz.* applying the techniques from [14] to the specific setting of group lasso with overlapping groups. We describe the theory and reasoning behind this approach. In Section 2.2.3 we derive bounds on the number of random i.i.d. Gaussian measurements needed to be taken for exact recovery of group sparse signals. We further derive bounds for the number of measurements required for robust recovery of signals as well. Section 2.2.4 outlines the experiments we performed and the corresponding results obtained. We conclude our chapter in Section 3.7.

Notations

We first introduce notations that we will use for the rest of the section. Consider a signal of length p, that is s sparse. Note here that in case of multidimensional signals like images, we assume they are vectorized to have length p. The coefficients of the signal are grouped into sets $\{G_i\}_{i=1}^{M}$, such that $\forall i \in \{1, 2, \cdots, M\}, G_i \subset \{1, 2, \cdots, p\}$. We denote the set of groups by $\mathcal{G} = \{G_i\}_{i=1..M}$, and $|\cdot|$ denotes the cardinality of a set. We let $x^\star$ be the (sparse) signal to be recovered, whose non zero coefficients lie in k of the M groups $\mathcal{G}^\star \subset \mathcal{G}$. Formally,

$$\mathcal{G}^\star = \{G_i \in \mathcal{G}^\star : \text{supp}(x^\star) \cap G_i \neq 0\}$$

We assume $|\mathcal{G}^\star| = k \leq M = |\mathcal{G}|$. We let $\mathbf{\Phi}_{n \times p}$ be a measurement matrix consisting of i.i.d. Gaussian entries of mean 0 and unit variance so that every column is a realization of an i.i.d. Gaussian length n vector with covariance I. For any vector $\boldsymbol{x} \in \mathbb{R}^p$, we denote by $\boldsymbol{x}_G$ a vector in $\mathbb{R}^p$ such that $(\boldsymbol{x}_G)_i = \boldsymbol{x}_i$ if $i \in G$, and 0 otherwise. We denote the observed vector by $\boldsymbol{y} \in \mathbb{R}^n \ : \boldsymbol{y} = \mathbf{\Phi}\boldsymbol{x}^\star$. The absence of a subscript following a norm $\|\cdot\|$ implies the ℓ_2 norm. The dual norm of $\|\cdot\|_p$ is denoted by $\|\cdot\|_p^*$. The convex hull of a set of points S is denoted by $\text{conv}(S)$.

2.2.2 Preliminaries

In this section, we will set up the problem that we wish to solve in this chapter. We will argue as to why exact recovery of the signal corresponds to the minimization of the atomic norm of the signal, with the atoms obeying certain properties governed by the signal structure.

Atoms and the atomic set

To begin with, let us (re)formalize the notion of atoms and the atomic norm of a signal (or vector), and in the process revisit some definitions from Chapter 1. We will restrict our attention to group-sparse signals in $\mathbb{R}^p$, though the same concepts can be extended to other spaces as well. Recall that we assume $\boldsymbol{x} \in \mathbb{R}^p$ can be decomposed as :

$$\boldsymbol{x} = \sum_{i=1}^{k} c_i \boldsymbol{a}_i, \quad c_i \geq 0$$

Where $\boldsymbol{a} \in \mathcal{A}$ are the atoms. Note that the sum notation, rather than the integral notation, implies that only a countable number of coefficients can be non-zero. As explained in Chapter 1, to obtain a "simple" representation of a vector, we look to minimize the atomic norm subject to constraints (equation (2.1)):

$$\hat{\boldsymbol{x}} = \underset{\boldsymbol{x} \in \mathbb{R}^p}{\operatorname{argmin}} \|\boldsymbol{x}\|_{\mathcal{A}} \ \textbf{s.t. } \boldsymbol{y} = \boldsymbol{\Phi x} \tag{2.1}$$

Assuming we are aware of the group structure $\mathcal{G}$, we now proceed to define the atomic set and the corresponding atomic norm for our framework:

$$\forall G \in \mathcal{G}, \text{ let}$$
$$A_G = \{a^G \in \mathbb{R}^p : ||(\boldsymbol{a}^G)_G||_2 = 1, (\boldsymbol{a}^G)_{G^c} = 0\}$$

$$\mathcal{A} = \{A_G\}_{G \in \mathcal{G}} \tag{2.2}$$

We now show that the atomic norm of a vector $x \in \mathbb{R}^p$ under the atomic set defined in equation (2.2) is equivalent to the overlapping group lasso norm defined in [37], a special case of which

is the standard group lasso norm [91]. Thus, minimizing the atomic norm in this case is exactly the same as the group lasso with overlapping groups.

Lemma 2.2.1. *Given any arbitrary set of groups $\mathcal{G}$, we have*

$$||\boldsymbol{x}||_{\mathcal{A}} = \Omega^{\mathcal{G}}_{overlap}(\boldsymbol{x})$$

where $\Omega^{\mathcal{G}}_{overlap}(\boldsymbol{x})$ is the overlapping group lasso norm defined in [37].

Proof. In (3.6), we can substitute $\boldsymbol{v}_G = c_G\boldsymbol{a}$, giving us $c_G = |c_G| \cdot ||a|| = ||c_G\boldsymbol{a}|| = \|\boldsymbol{v}_G\|$. Hence,

$$\begin{aligned}
||\boldsymbol{x}||_{\mathcal{A}} &= \inf\left\{\sum_{\boldsymbol{a}\in\mathcal{A}} c_a : \boldsymbol{x} = \sum_{\boldsymbol{a}\in\mathcal{A}} c_a\boldsymbol{a} \;\; c_a \geq 0 \;\; \forall \boldsymbol{a} \in \mathcal{A}\right\} \\
&= \inf\left\{\sum_{G\in\mathcal{G}} ||v_G|| \;\; : \boldsymbol{x} = \sum_{G\in\mathcal{G}} \boldsymbol{v}_G\right\} \\
&= \Omega^{\mathcal{G}}_{overlap}(\boldsymbol{x})
\end{aligned}$$

□

Corollary 2.2.2. *Under the atomic set defined in (2.2) , when $\mathcal{G}$ partitions $\mathbb{R}^p$,*

$$||\boldsymbol{x}||_{\mathcal{A}} = \sum_{G\in\mathcal{G}} ||\boldsymbol{x}_G||$$

Proof. $\Omega^{\mathcal{G}}_{overlap} = \sum_{G\in\mathcal{G}} ||\boldsymbol{x}_G||$ in the non overlapping case. □

Thus, (2.1) yields:

$$\hat{x} = \underset{\boldsymbol{x}\in\mathbb{R}^p}{\operatorname{argmin}} \;\; \Omega^{\mathcal{G}}_{overlap}(\boldsymbol{x}) \;\; \textbf{s.t. } \boldsymbol{y} = \boldsymbol{\Phi x} \tag{2.3}$$

which can be solved using the replication based method outlined in Chapter 1.

Also note that we can directly compute the dual of the atomic norm from the set of atoms

$$\|\boldsymbol{u}\|_{\mathcal{A}}^{*} = \sup_{\boldsymbol{a}\in\mathcal{A}} \langle \boldsymbol{a}, \boldsymbol{u}\rangle = \max_{G\in\mathcal{G}} ||\boldsymbol{u}_G|| \tag{2.4}$$

The dual norm will be useful in our derivations below.

Gaussian Widths and Exact Recovery

Following [14], we define the *tangent cone* and *normal cone* at $\boldsymbol{x}^\star$ with respect to $conv(\mathcal{A})$ under $||\boldsymbol{x}||_{\mathcal{A}}$ as [72]:

$$\mathcal{T}_{\mathcal{A}}(\boldsymbol{x}^\star) = \text{cone}\{\boldsymbol{z} - \boldsymbol{x}^\star \ \ : ||\boldsymbol{z}||_{\mathcal{A}} \leq ||\boldsymbol{x}^\star||_{\mathcal{A}}\} \tag{2.5}$$

$$\begin{aligned}\mathcal{N}_{\mathcal{A}}(\boldsymbol{x}^\star) &= \{\boldsymbol{u} \ : \ \langle \boldsymbol{u}, \boldsymbol{z}\rangle \leq 0, \ \ \forall \boldsymbol{z} \in \mathcal{T}_{\mathcal{A}}(\boldsymbol{x}^\star)\} \\ &= \{\boldsymbol{u} \ : \ \langle \boldsymbol{u}, \boldsymbol{x}^\star\rangle = t\|\boldsymbol{x}\|_{\mathcal{A}} \\ &\quad \text{and } \|\boldsymbol{u}\|_{\mathcal{A}}^{*} \leq t \text{ for some } t \geq 0\}\end{aligned} \tag{2.6}$$

We note that, from [14] (Prop. 2.1), $\hat{x} = x^\star$ (2.1) is unique *iff*

$$null(\boldsymbol{\Phi}) \cap \mathcal{T}_{\mathcal{A}}(\boldsymbol{x}^\star) = \{\boldsymbol{0}\} \tag{2.7}$$

Hence, we require that the tangent cone at $\boldsymbol{x}^\star$ intersects the nullspace of $\boldsymbol{\Phi}$ only at the origin, to guarantee exact recovery.

Before we state the main recovery result from [14], we define the *Gaussian width* of a set:

Definition Let $\mathbb{S}^{p-1}$ denote the unit sphere in $\mathbb{R}^p$. The Gaussian width $\omega(S)$ of a set $S \in \mathbb{S}^{p-1}$

is

$$\omega(S) = \mathbb{E}_{\boldsymbol{g}} \left[\sup_{\boldsymbol{z} \in S} \boldsymbol{g}^T \boldsymbol{z} \right]$$

where $\boldsymbol{g} \sim \mathcal{N}(0, I)$

Gordon uses the Gaussian width to provide bounds on the probability that a random subspace of a certain dimension misses a subset of the sphere [33]. In [14], these results are specialized to the case of atomic norm recovery. In particular, we will make use of the following:

Proposition 2.2.3. *[[14], Corollary 3.2] Let $\Phi : \mathbb{R}^p \to \mathbb{R}^n$ be a random map with i.i.d. zero-mean Gaussian entries having variance $1/n$. Further let $\Omega = T_{\mathcal{A}}(x^*) \cap \mathbb{S}^{p-1}$ denote the spherical part of the tangent cone $T_{\mathcal{A}}(\boldsymbol{x}^\star)$. Suppose that we have measurements $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x}^\star$, and we solve the convex program (2.1). Then $\boldsymbol{x}^\star$ is the unique optimum of (2.1) with high probability provided that*

$$n \geq \omega(\Omega)^2 + \mathcal{O}(1).$$

To complete our problem setup we will also restate Proposition 3.6 in [14] :

Proposition 2.2.4. *(Proposition 3.6 in [14]) Let C be any non-empty convex cone in $\mathbb{R}^p$, and let $g \sim \mathcal{N}(0, I)$ be a Gaussian vector. Then:*

$$\omega(C \cap \mathbb{S}^{p-1}) \leq \mathbb{E}_g[\mathrm{dist}(g, C^*)] \tag{2.8}$$

where $\mathrm{dist}(.,.)$ denotes the Euclidean distance between a point and a set, and C^* is the dual cone of C

We can then square (2.8) use Jensen's inequality to obtain

$$\omega(C \cap \mathbb{S}^{p-1})^2 \leq \mathbb{E}_g[\mathrm{dist}(g, C^*)^2] \tag{2.9}$$

We note here that the dual cone of the tangent cone is the normal cone, and vice-versa.

Thus, to derive measurement bounds, we only need to calculate the square of the Gaussian width of the intersection of the tangent cone at $x^\star$ with respect to the atomic norm and the unit sphere. This value can be bounded by the distance of a Gaussian random vector to the normal cone at the same point, as implied by (2.9). In the next section, we derive bounds on this quantity.

2.2.3 Gaussian Width of the Normal Cone of the Group Sparsity Norm

For generic groups $\mathcal{G}$, we have

$$\begin{aligned} &v \in \mathcal{N}_{\mathcal{A}}(x^\star) \Leftrightarrow \exists \gamma \geq 0 \quad : \langle v, x^\star \rangle = \gamma \|x^\star\|_{\mathcal{A}}, \\ &\|v_G\| = \gamma \text{ if } G \in \mathcal{G}^\star, \;\; \|v_G\| \leq \gamma \text{ if } G \notin \mathcal{G}^\star. \end{aligned} \tag{2.10}$$

It is not hard to see that, in the case of disjoint groups,

$$\begin{aligned} \mathcal{N}_{\mathcal{A}}(x^\star) = \{z \in \mathbb{R}^p : z_i = \gamma \frac{(x^\star)_i}{||x^\star_G||} \;\; \forall G \in \mathcal{G}^\star, \\ ||z_G|| \leq \gamma \;\; \forall G \;\notin\; \mathcal{G}^\star, \gamma \geq 0\} \end{aligned} \tag{2.11}$$

However, in the case of overlapping groups, no such closed form exists.

We now prove the main result of this chapter, a sufficient number of Gaussian measurements needed to recover a group-sparse signal:

Theorem 2.2.5. *To exactly recover a k-group sparse signal decomposed into M groups in $\mathbb{R}^p$, $(\sqrt{2\log(M-k)} + \sqrt{B})^2 k + kB$ i.i.d. Gaussian measurements are sufficient.*

We defer the proof of this result to Appendix A.1

If the groups are disjoint to begin with, the normal cone will be given by (2.11), and $\|v_S\|^2 = k$. Also, in this case, we have $|S| = kB$. We see that we do not pay an additional penalty when the groups overlap. This fact is surprising, since one would expect that one would need more measurements to effectively capture the dependencies among the overlapping groups.

Remarks

The kB term in the bound is an upper-bound on the signal sparsity. In the case of highly overlapping groups, this value may be much larger than the signal sparsity, but such cases seldom arise in real-world applications. If the group sizes are vastly different, then it is pessimistic to bound the quantity with the maximum group size B, but this yields a simple expression for the measurements needed. It is of course possible to obtain tighter bounds using the techniques in our work for cases where the groups are of varying sizes.

It can be seen from Theorem 2.2.5 that the number of measurements is linear in k and B. Hence, the number of measurements that are sufficient for signal recovery grows linearly with the number of active groups in the signal, and also the maximum group size. This can be seen analogous to the linear dependence of the lasso bound on the sparsity s of the signal, though for overlapping groups, $kB \neq s$.

We note that although we pay no extra price to measure the signal when the groups overlap, there is an additional cost in the recovery process of the signal, in that the groups need to first be separated by replication of the coefficients [37], or resort to a primal-dual method to solve the problem [56].

Finally, we compare the bound we obtain to the standard lasso measurement bound [14]:

$$(2s+1)\log(p-s) \tag{2.12}$$

The bound we obtain in Theorem 2.2.5 can be upper bounded by

$$2k\max\{2\log(M), B\} + kB \tag{2.13}$$

Noting that $s \leq kB$ with equality when the groups do not overlap. In this case, (2.13) evaluates to

$$\begin{aligned} &\frac{2s}{B}\max\{2\log(M), B\} + s \\ &= (2s+1)\frac{\max\{2\log(M), B\}}{B} \end{aligned}$$

which is smaller than the lasso bound (2.12) by a factor of roughly $\frac{\log(M)}{B\log(p)}$. So, in most cases, our bound shows that the we can perform better than the conventional lasso by exploiting the additional group structured information that is available.

Noisy Observations

The results we obtain can be easily extended to the case where we obtain noisy observations, assuming that the noise is bounded. In the noisy case, we observe

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x}^\star + \theta, \;\; \|\theta\| \leq \delta$$

We then solve the atomic norm minimization problem, with a relaxed constraint to take into account the bounded noise:

$$\hat{\boldsymbol{x}} = \underset{\boldsymbol{x}\in\mathbb{R}^p}{\operatorname{argmin}} ||\boldsymbol{x}||_{\mathcal{A}} \ \textbf{s.t.} \ \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}\| \leq \delta \tag{2.14}$$

We restate corollary 3.3 from [14]:

Proposition 2.2.6. *[[14], Corollary 3.3] Let $\Phi : \mathbb{R}^p \to \mathbb{R}^n$ be a random map with i.i.d. zero-mean Gaussian entries having variance $1/n$. Further let $\Omega = T_{\mathcal{A}}(x^*) \cap \mathbb{S}^{p-1}$ denote the spherical part of the tangent cone $T_{\mathcal{A}}(x^\star)$. Suppose that we have measurements $y = \Phi x^\star + \theta$, and $\|\theta\| \leq \delta$. Suppose we solve the convex program (2.14). Let $\hat{x}$ denote the optimum of (2.14). Also, suppose $\|\Phi z\| \geq \epsilon\|z\| \ \ \forall z \in T_{\mathcal{A}}(x^\star)$. Then $\|x^\star - \hat{x}\| \leq \frac{2\delta}{\epsilon}$ with high probability provided that*

$$n \geq \frac{\omega(\Omega)^2}{(1-\epsilon)^2} + \mathcal{O}(1).$$

Substituting the result in Theorem 2.2.5 in Proposition 2.2.6, we have the following corollary yielding a sufficient condition to accurately recover a signal when the measurements are corrupted with bounded noise:

Corollary 2.2.7. *Suppose we wish to recover a $k-$ group sparse signal having M groups, such that the maximum group size is B. Let $\hat{x}$ be the optimum of the convex program (2.14). To have $\|\hat{x} - x^\star\| \leq \frac{2\delta}{\epsilon}$ with high probability,*

$$\frac{(\sqrt{2\log(M-k)} + \sqrt{B})^2 k + kB}{(1-\epsilon)^2}$$

i.i.d. Gaussian measurements are sufficient.

2.2.4 Experiments and Results

We extensively tested our method against the standard lasso procedure. In the case where the groups overlap, we use the replication method outlined in [37], to reduce the optimization problem to that of non overlapping groups. We compare the number of measurements needed for our method with that needed for the lasso. For the lasso, it would be instructive to keep in mind the bound derived in [14] , *viz.* $(2s + 1)\log(p - s)$. In the case of non overlapping groups, the bound evaluates to $(2kB + 1)\log(kM - kB)$. We generate length $p = 2000$ signals, made up of $M = 100$ non-overlapping groups of size $B = 20$. We set $k = 5$ groups to be "active", and the values within the groups are drawn from a uniform $[0, 1]$ distribution. The active groups are assigned uniformly at random. The sparsity of the signal will be $s = 100$

We use SpaRSA [90] for the lasso and the group lasso with overlap, learning λ over a grid. Figure 5 displays the mean reconstruction error $||\hat{x} - x^*||_2^2/p$ as a function of the number of random measurements taken. The errors have been averaged over 100 tests, and each time a new random signal was generated with the above mentioned parameters.

From the parameters considered, we conclude that ≈ 380 measurements are sufficient to recover the signal. When we have 380 measurements, the lasso does not recover the signal exactly, as seen in Figure 5.

To show that the bound we compute holds regardless of the complexity of groupings, we consider the following scenario: Suppose we have $M = 100$ groups, each of size $B = 40$. $k = 5$ of those groups are active, and the values within each group are assigned from a uniform $[-1, 1]$ distribution. We arrange these groups in three configurations:

1. The groups do not overlap, yielding a signal of length $p = 4000$, and signal sparsity $s = 200$.

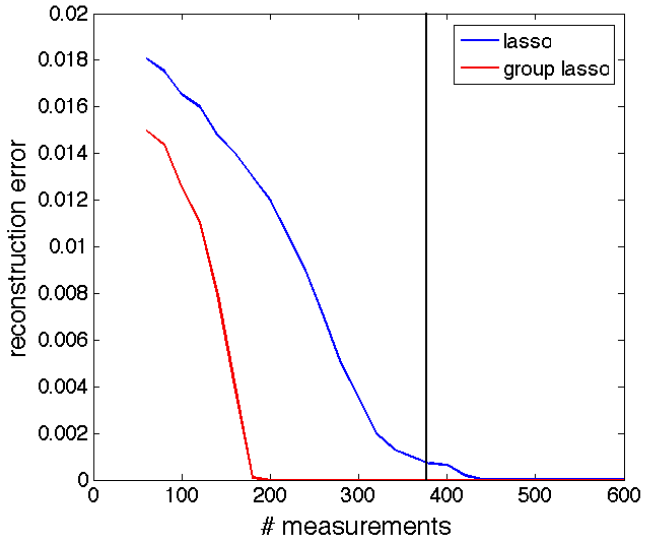


Figure 5: The group lasso (red) compared with the lasso (blue). The vertical line indicates our bound. Note that our bound (380) predicts exact recovery of the signal, while at the same value, the lasso does not recover the signal

2. A partial overlapping scenario, where apart from the first and last group, every group has 20 elements in common with a group above it, and 20 common with the group below, giving $p = 2020$, $s \in [120, \ 200]$ depending on which of the 100 groups are active.

3. An almost complete overlap, where apart from one element in each group, the remaining elements are common to each group. This leads to $p = 139$ and $s = 44$

4. We also considered cases intermediate to the ones listed above. Specifically, we considered (a) a highly overlapping scenario which is identical to the previous case, but with odd and even groups disjoint, giving $p = 178$ and $s \leq 80$. We also consider (b) a random overlap case where the first 50 groups are non overlapping and the remaining 50 are assigned uniformly at random from the existing $p = 2000$ indices. $s \leq 200$ in this case.

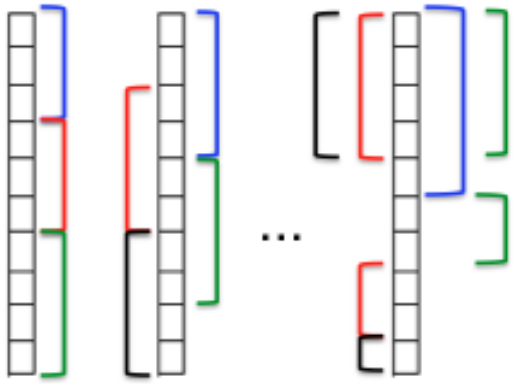
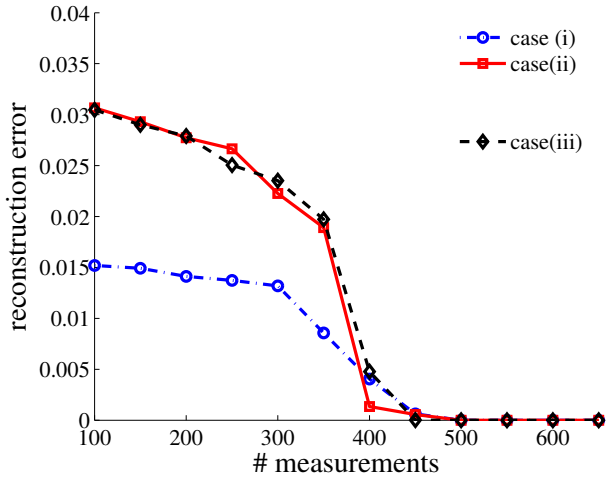


Figure 6: Types of groupings considered. Each set of coefficients encompassed by one color belongs to one group.

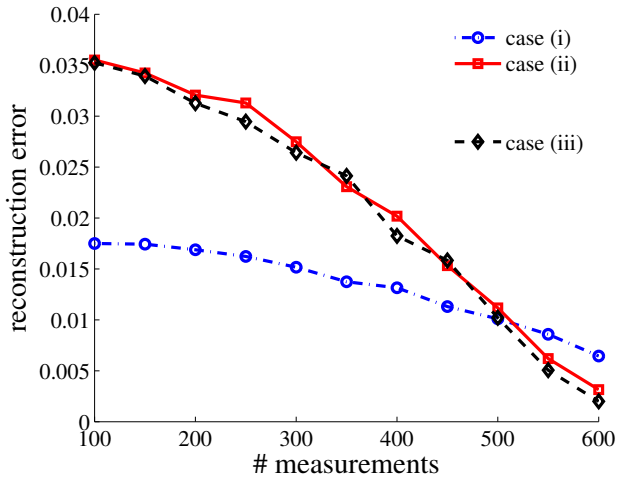
The scenarios we consider are depicted in Figure 6. In each of the cases, we compute the bound to be ≈ 630. The bound becomes looser as the complexity of the groupings increases. This, as argued before, is a result of the bound for the signal sparsity becoming looser.

We can see from Figure 7a that our group lasso bound (≈ 630) holds for all cases. For the sake of comparison, we considered the lasso performance on the signals in cases (i) - (iv) as well, and these are plotted in Figure 7b. From the values of p and s computed for the four cases, we have the corresponding bounds for the lasso [14] to be 3305 for the no overlap case (i), [1819, 3010] for the partial overlap case (ii) and 405 for the almost complete overlap case (iii) respectively. The lasso bounds for case (iv a) and (iv b) are 738 and 3000 respectively. Another thing to note is that, apart from cases (iii) and (iv a), the group lasso always outperforms the lasso. This leads us to believe that when there is excessive overlap between groups, the knowledge of the group structure does not aid in signal reconstruction.

Our final experiment outlines the relationship between the number of groups M and the number of measurements needed, when $k = \frac{M}{10}$. We consider the partial overlap scenario as mentioned before in case (ii), with $B = 10$. Figure 8 shows that as we increase the number of



(a) performance of the group lasso on cases considered in Figure 6. Note that our bound evaluates to 630, clearly sufficient measurements to recover the signal in all cases.



(b) performance of the lasso on cases considered in Figure 6.

Figure 7: (Best seen in color) Performance on various grouping schemes. The group lasso outperforms the lasso in all cases apart from (iii) and (iv a). This shows that as the amount of overlap increases, the group lasso does not yield any advantage as compared to the lasso, and if anything, performs worse.

total groups, we naturally need more measurements. It is also instructive to note that since the number of active groups is proportional to M, we get an almost linear relationship between M and the number of measurements needed for perfect recovery. This effect is captured in our bound, which scales linearly with k, the number of active groups, which is linear in M, the total number of groups in this experiment. The probability of error is computed empirically from 100 runs for each $(measurement, M)$ pair.

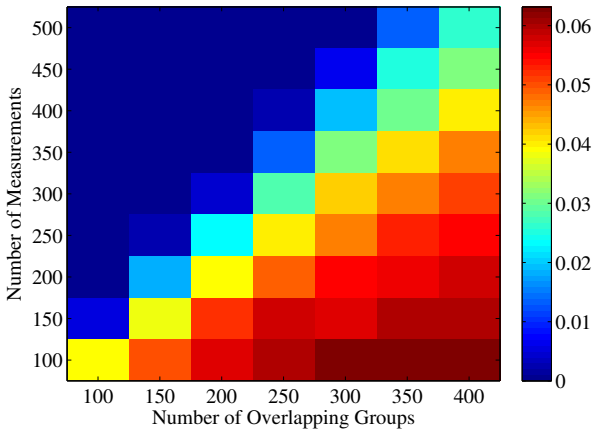


Figure 8: Number of measurements needed *vs* the total number of groups for recovery. The image shows the probability of error, with blue indicating values that are (nearly) zero. The maximum value on the plot corresponds to a 0.06 probability of error. (Best seen in color).

Another thing to note with regards to Figure 8 is that the x-axis shows the number of groups in the signal, since our bound depends only on that. In the present setup, the corresponding dimensionality of the signal is $(505, 755, 1005, 1255, 1505, 1755, 2005)$ respectively for each M in Figure 8.

2.2.5 Conclusion

We showed that, when additional structure about the support of the signal to be estimated is known, we can recover the signal in much fewer measurements that what would be needed

in the standard compressed sensing framework using the lasso. Also, we showed that we surprisingly do not pay an extra penalty when the groups overlap each other. Moreover, the bound holds for arbitrary group structures, and can be used in a variety of applications. The bounds we derive are tight, and can be extended to subGaussian measurement matrices by incurring a constant penalty. Experimental results agree with the bounds we obtained.

2.3 The Sparse Overlapping Sets Lasso

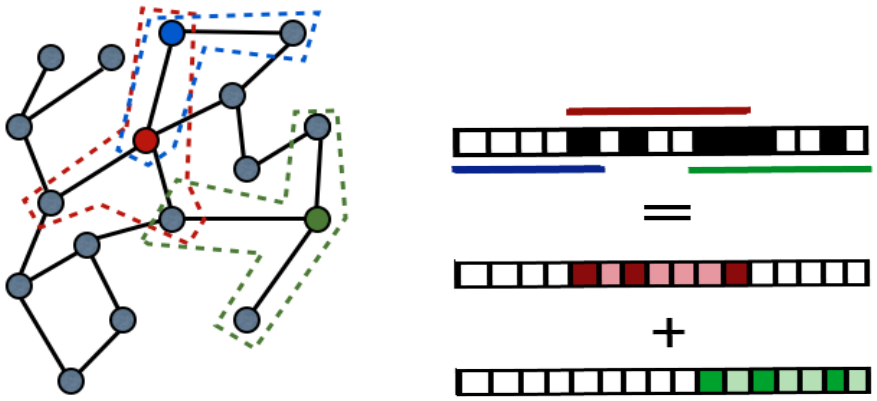
In the previous section, we derived sample complexity bounds for the group lasso with overlapping groups. In this section, we extend the overlapping group lasso to select a small number of overlapping groups, when the groups themselves exhibit sparsity within themselves.

2.3.1 Introduction

Binary logistic regression plays a major role in many machine learning and signal processing applications. In modern applications where the number of features far exceeds the number of observations, one typically enforces the solution to contain only a few non zeros. The lasso [83] is a commonly used method to constrain the solution (henceforth also referred to as the coefficients) to be sparse. The notion of sparsity leads to more interpretable solutions in high dimensional machine learning applications, and has been extensively studied in [3, 10, 59, 65], among others. In cases when we have additional information about the structure within the non zero coefficients, we saw in the previous section how one can use the group lasso to recover the signals of interest.

The group lasso forces all the coefficients in a group to be active at once: if a coefficient is selected for the task at hand, then all the coefficients in that group are selected. This is the case even when the groups overlap.

While the group lasso has enjoyed tremendous success in high dimensional feature selection applications, we are interested in a much less restrictive form of structured feature selection for classification. Suppose that the features can be arranged into *overlapping* groups based on some notion of similarity, depending on the application. For example, in Figure 9a, the features can be organized into a graph (similar features being connected), and each feature



(a) Grouping features based on similarity. Vertices denote coefficients and edges indicate the similarity of the corresponding features. Each group (indicated by dashed colored lines) corresponds to a particular vertex (highlighted by the same color), and its neighbors. The groups overlap, and if a vertex is selected, it should encourage selection its neighbors.

(b) Decomposition of coefficient vector into overlapping sparse groups. The vector can be decomposed as a sum of two vectors, each corresponding to a single group (green and red). The lighter shaded regions in the right hand side correspond to zeros in the active groups, and the darker regions correspond to non zero coefficients. The vertical bars in the LHS indicate various overlapping groups

Figure 9: (Best seen in color) Grouping features based on similarity (a) and its decomposition in terms of sparse overlapping sets (b). The sparse vector that determines what features are selected takes into account the groups formed due to the graph in (a)

forms a group with its neighbors. The notion of similarity can be loosely defined, and only suggests that if a feature is relevant for the learning task at hand, then features similar to it may also be relevant. It is known that while many features may be similar to each other, not all similar features are relevant for the specific learning problem. Figure 9b illustrates the pattern we are interested in. In such a setting, we want to select [1] similar features (i.e., groups), but only a (sparse) subset of the features in the (selected) groups may themselves be selected. We propose a new procedure called Sparse Overlapping Sets (SOS) lasso to reflect this situation in the coefficients recovered.

[1] A feature or group of features is “selected" if its corresponding regression coefficient(s) is non zero

As an example, consider the task of identifying relevant genes that play a role in predicting a disease. Genes are organized into pathways [80], but not every gene in a pathway might be relevant for prediction. At the same time, it is reasonable to assume that if a gene from a particular pathway is relevant, then other genes from the same pathway may also be relevant. In such applications, the group lasso may be too constraining while the lasso may be too under-constrained.

A major motivating factor for our approach comes from multitask learning. Multitask learning can be effective when features useful in one task are also useful for other tasks, and the group lasso is a standard method for selecting a common subset of features [49]. In this section, we consider the case where (1) the available features can be organized into groups according to a notion of similarity and (2) features useful in one task are similar, but not necessarily identical, to the features best suited for other tasks. Later in the chapter, we apply this idea to multi-subject fMRI prediction problems.

2.3.2 Past Work

When the groups of features do not overlap, [76] proposed the Sparse Group Lasso (SGL) to recover coefficients that are both within- and across- group sparse. SGL and its variants for multitask learning has found applications in character recognition [78, 79], climate and oceanology applications [15], and in gene selection in computational biology [76]. In [43], the authors extended the method to handle tree structured sparsity patterns, and showed that the resulting optimization problem admits an efficient implementation in terms of proximal point operators. Along related lines, the exclusive lasso [94] can be used when it is explicitly known that features in certain groups are negatively correlated. When the groups overlap,

[37, 63] proposed a modification of the group lasso penalty so that the resulting coefficients can be expressed as a union of groups. They proposed a replication-based strategy for solving the problem, which has since found application in computational biology [37] and image processing [67], among others. The authors in [57] proposed a method to solve the same problem in a primal-dual framework, that does not require coefficient replication. Risk bounds for problems with structured sparsity inducing penalties (including the lasso and group lasso) were obtained by [51] using Rademacher complexities. Sample complexity bounds for model selection in linear regression using the group lasso (with possibly overlapping groups) also exist [68]. The results naturally hold for the standard group lasso [91], since non overlapping groups are a special case.

For logistic regression, [3, 10, 59, 65] and references therein have extensively characterized the sample complexity of identifying the correct model using ℓ_1 regularized optimization. In [65], the authors introduced a new optimization framework to solve the logistic regression problem: minimize [2] a linear cost function subject to a constraint on the ℓ_1 norm of the solution.

2.3.3 Our Contributions

In this section, we consider an optimization problem of the form in [65], but for coefficients that can be expressed as a union of overlapping groups. Not only are only a few groups selected, but the selected groups themselves are also sparse. In this sense, our constraint can be seen as an extension of SGL [76] for overlapping groups where the sparsity pattern lies in a union of groups. We are mainly interested in classification problems, but the method can also be applied to regression settings, by making an appropriate change in the loss function

[2] The authors in [65] write the problem as a maximization. We minimize the negative of the same function

of course. We consider a union-of-groups formulation as in [37], but with an additional sparsity constraint on the selected groups. To this end, we analyze the Sparse Overlapping Sets (SOS) lasso, where the overlapping sets might correspond to coefficients of features arbitrarily grouped according to the notion of similarity.

We introduce a constraint that promotes sparsity patterns that can be expressed as a union of sparsely activated groups. The main contribution of this section is a theoretical analysis of the consistency of the SOSlasso estimator, under a logistic regression setting. Based on certain parameter settings, our method reduces to other known cases of penalization for sparse high dimensional recovery. Specifically, our method generalizes the group lasso for logistic regression [37, 53], and also extends to handle groups that can arbitrarily overlap with each other. We also recover results for the lasso for logistic regression [3, 10, 59, 65]. In this sense, our work unifies the lasso, the group lasso as well as the sparse group lasso for logistic regression to handle overlapping groups. To the best of our knowledge, this is the first chapter that provides such a unified theory and sample complexity bounds for all these methods.

In the case of linear regression and multitask learning, our work generalizes the work of [78, 79], where the authors consider a similar situation with non overlapping subsets of features. We assume that the features can arbitrarily overlap. When the groups overlap, the methods mentioned above suffer from a drawback: entire groups are set to zero, in effect zeroing out many coefficients that might be relevant to the tasks at hand. This has undesirable effects in many applications of interest, and the authors in [37] propose a version of the group lasso to circumvent this issue.

We also test our regularizer on both toy and real datasets. Our experiments reinforce our theoretical results, and demonstrate the advantages of the SOSlasso over standard lasso and group lasso methods, when the features can indeed be grouped according to some notion of

similarity. We show that the SOSlasso is especially useful in multitask Functional Magnetic Resonance Imaging (fMRI) applications, and gene selection applications in computational biology in Chapter 4

To summarize, the main contributions of this section are the following:

1. **New regularizers for structured sparsity:** We propose the Sparse Overlapping Sets (SOS) lasso, a convex optimization problem that encourages the selection of coefficients that are both within-and across- group sparse. The groups can arbitrarily overlap, and the pattern obtained can be represented as a union of a small number of groups. This generalizes other known methods, and provides a common regularizer that can be used for any structured sparse problem with two levels of hierarchy [3]: groups at a higher level, and singletons at the lower level.

2. **New theory for logistic regression with structured sparsity:** We provide a theoretical analysis for the consistency of the SOSlasso estimator, under the logistic observation model. The general results we obtain specialize to the lasso, the group lasso (with or without overlapping groups) and the sparse group lasso. We obtain a bound on the sample complexity of the SOSlasso under both independent and correlated Gaussian measurement designs, and this in turn also translates to corresponding results for the lasso and the group lasso. In this sense, we obtain a unified theory for performing structured variable selection in high dimensions.

[3] Further levels can also be added as in [43], but that is beyond the scope of this chapter.

2.3.4 Logistic Regression with Structured Sparsity

In this section, we formalize our problem. We first describe the notations that we use in the sequel. Uppercase and lowercase bold letters indicate matrices and vectors respectively. We assume a sparse learning framework, with a feature matrix $\boldsymbol{\Phi} \in \mathbb{R}^{n \times p}$. We assume each element of $\boldsymbol{\Phi}$ to be distributed as a standard Gaussian random variable. Assuming the data to arise from a Gaussian distribution simplifies analysis, and allows us to leverage tools from existing literature. Later in the section, we will allow for correlations in the features as well, reflecting a more realistic setting. In the results that follow, C is a constant, the value of which can be different from one result to the other.

We focus on classification, and assume the following logistic regression model. Each observation $\boldsymbol{y}_i \in \{-1, +1\}, \;\; i = 1, 2, \ldots, n$ is randomly distributed according to the logistic model

$$\mathbb{P}\left(\boldsymbol{y}_i = 1\right) = f(\langle \boldsymbol{\phi}_i, \boldsymbol{x}^\star \rangle) \tag{2.15}$$

where $\boldsymbol{\phi}_i$ is the i^{th} row of $\boldsymbol{\Phi}$, and $\boldsymbol{x}^\star \in \mathbb{R}^p$ is the (unknown) coefficient vector of interest in our setting.

$$f(z) = \frac{\exp(z)}{1 + \exp(z)}$$

where z is a scalar.

The coefficient vector of interest is assumed to have a special structure. Specifically, we assume that $\boldsymbol{x} \in \mathcal{C} \subset B_2^p$, where B_2^p is the unit ball in $\mathbb{R}^p$. This motivates the following optimization problem [65]:

$$\widehat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \sum_{i=1}^{n} -\boldsymbol{y}_i \left\langle \boldsymbol{\phi}_i, \boldsymbol{x} \right\rangle \;\; \textbf{s.t.} \;\; \boldsymbol{x} \in \mathcal{C} \tag{2.16}$$

The statistical accuracy of $\widehat{x}$ can be characterized in terms of the *mean width* of $\mathcal{C}$, which is defined as follows

Definition Let $\boldsymbol{g} \in \mathcal{N}(0, \boldsymbol{I})$. The mean width of a set $\mathcal{C}$ is defined as

$$\omega(\mathcal{C}) = \mathbb{E}_{\boldsymbol{g}} \left[\sup_{\boldsymbol{x} \in \mathcal{C} - \mathcal{C}} \langle \boldsymbol{x}, \boldsymbol{g} \rangle \right]$$

where $\mathcal{C} - \mathcal{C}$ denotes the Minkowski set difference.

Note that definition 2.3.4 is equivalent to the definition of the mean width in the previous section for the atomic set. In most cases of practical importance, the set we are interested in will be symmetric about the origin, and hence we can replace the Minkowski difference by the set itself.

The next result follows immediately from Theorem 1.1, Corollary 1.2, and Corollary 3.3 of [65].

Theorem 2.3.1. *Let $\boldsymbol{\Phi} \in \mathbb{R}^{n \times p}$ be a matrix with i.i.d. standard Gaussian entries, and let $\mathcal{C} \subset B_2^p$. Assume $\frac{\boldsymbol{x}^\star}{\|\boldsymbol{x}^\star\|_2} \in \mathcal{C}$, and the observations follow the model (2.15) above. Let $\delta > 0$, and suppose*

$$n \geq C\delta^{-2}\omega(\mathcal{C})^2$$

Then, with probability at least $1 - 8\exp(-c\delta^2 n)$, the solution $\widehat{\boldsymbol{x}}$ to the problem (2.16) satisfies

$$\left\| \widehat{\boldsymbol{x}} - \frac{\boldsymbol{x}^\star}{\|\boldsymbol{x}^\star\|_2} \right\|_2^2 \leq \delta \max(\|\boldsymbol{x}^\star\|^{-1}, 1)$$

where C, c are positive constants.

In this section, we construct a new penalty that produces a convex set $\mathcal{C}$ that encourages

structured sparsity in the solution of (2.16). We show that the resulting optimization can be efficiently solved. We bound the mean width of the set, which yields new bounds for logistic regression with structured sparsity, via Theorem 2.3.1.

A New Penalty for Structured Sparsity

We are interested in the following form of structured sparsity. Assume that the features can be organized into *overlapping* groups based on a user-defined measure of similarity, depending on the application. Moreover, assume that if a certain feature is relevant for the learning task at hand, then features similar to it may also be relevant. These assumptions suggest a structured pattern of sparsity in the coefficients wherein a subset of the groups are relevant to the learning task, and within the relevant groups a subset of the features are selected. In other words, $\boldsymbol{x}^\star \in \mathbb{R}^p$ has the following structure:

- its support is localized to a union of a subset of the groups, and
- its support is localized to a sparse subset within each such group

Assume that the features can be grouped according to similarity into M (possibly overlapping) groups $\mathcal{G} = \{G_1, G_2, \ldots, G_M\}$ and consider the following definition of structured sparsity.

Definition We say that a vector $\boldsymbol{x}$ is (k, α)-group sparse if $\boldsymbol{x}$ is supported on at most $k \leq M$ groups and at most a fraction $\alpha \in (0, 1]$ of the elements in the union of groups are non zero.

Note that $\alpha = 0$ corresponds to $\boldsymbol{x} = \mathbf{0}$.

To encourage such sparsity patterns we define the following penalty. Given a group $G \in \mathcal{G}$, we define the set

$$\mathcal{W}_G = \{\boldsymbol{w} \in \mathbb{R}^p : \ \boldsymbol{w}_i = 0 \ \textbf{if} \ i \notin G\}$$

We can then define

$$\mathcal{W}(\boldsymbol{x}) = \left\{ \boldsymbol{w}_{G_1} \in \mathcal{W}_{G_1},\ \ \boldsymbol{w}_{G_2} \in \mathcal{W}_{G_2}, \ldots, \boldsymbol{w}_{G_M} \in \mathcal{W}_{G_M} : \ \sum_{G \in \mathcal{G}} \boldsymbol{w}_G = \boldsymbol{x} \right\}$$

That is, each element of $\mathcal{W}(x)$ is a set of vectors, one from each $\mathcal{W}_G$, such that the vectors sum to $\boldsymbol{x}$. As shorthand, in the sequel we write $\{\boldsymbol{w}_G\} \in \mathcal{W}(\boldsymbol{x})$ to mean a set of vectors that form an element in $\mathcal{W}(\boldsymbol{x})$

For any $\boldsymbol{x} \in \mathbb{R}^p$, define

$$h(\boldsymbol{x}) \ := \ \inf_{\{\boldsymbol{w}_G\} \in \mathcal{W}(\boldsymbol{x})} \sum_{G \in \mathcal{G}} \left(\sqrt{B} \|\boldsymbol{w}_G\|_2 + \mu \|\boldsymbol{w}_G\|_1 \right) \tag{2.17}$$

The logistic *SOSlasso* is the optimization in (2.16) with $h(\boldsymbol{x})$ as defined in (2.17) determining the structure of the constraint set $\mathcal{C}$, and hence the form of the solution $\widehat{\boldsymbol{x}}$. The ℓ_2 penalty promotes the selection of only a subset of the groups, and the ℓ_1 penalty promotes the selection of only a subset of the features within a group.

Definition We say the set of vectors $\{\boldsymbol{w}_G\} \in \mathcal{W}(\boldsymbol{x})$ is an optimal representation of $\boldsymbol{x}$ if they achieve the $\inf$ in (2.17).

The objective function in (2.17) is convex and coercive. Hence, $\forall \boldsymbol{x}$, an optimal representation always exists.

We also define the “group support norm" as follows:

Definition Given a set of M groups $\mathcal{G}$, for any vector $\boldsymbol{x}$ and its optimal representation $\{\boldsymbol{w}_G\} \in \mathcal{W}(\boldsymbol{x})$, noting that $x = \sum_{G \in \mathcal{G}} \boldsymbol{w}_G$, define

$$\|\boldsymbol{x}\|_{\mathcal{G},0} = \sum_{G \in \mathcal{G}} \mathbb{1}_{\{\|\boldsymbol{w}_G\| \neq 0\}}$$

2.3.5 Analysis of the SOSlasso Penalty

Recall that we defined $h(\boldsymbol{x})$ as in (2.17). We can define the penalty in full generality, by defining constants $\alpha_G,\ \ \beta_G$ that tradeoff the ℓ_2 and ℓ_1 norm terms for each group:

$$h(\boldsymbol{x}) = \inf_{\{\boldsymbol{w}_G\} \in \mathcal{W}(\boldsymbol{x})} \sum_{G \in \mathcal{G}} \left(\alpha_G \sqrt{B} \|\boldsymbol{w}_G\|_2 + \beta_G \mu \|\boldsymbol{w}_G\|_1 \right) \tag{2.18}$$

This makes the model extremely flexible, and can be used to recover sparsity patterns that are structured in a very general sense of the word. However, we will restrict ourselves to the case considered in (2.17), for ease of exposition. All the results we derive can be extended to the general setting with $\alpha_G,\ \ \beta_G$, by including the appropriate constant terms where necessary.

Note that the sum of the terms $\mu\|\boldsymbol{w}_G\|_1$ does not yield the standard ℓ_1 norm of the vector $\boldsymbol{x}$, but instead an ℓ_1-like term that is made up of a weighted sum of the absolute value of the coefficients in the vector. The weight is proportional to the number of groups to which a coordinate belongs.

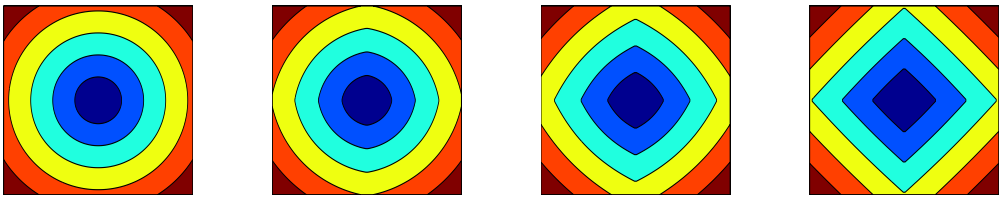
Remarks :

The SOSlasso penalty can be seen as a generalization of different penalty functions previously explored in the context of sparse linear and/or logistic regression:

- If each group in $\mathcal{G}$ is a singleton, then the SOSlasso penalty reduces to the standard ℓ_1 norm, and the problem reduces to the lasso for logistic regression [10, 83]
- if $\mu = 0$ in (2.17), then we are left with the latent group lasso [37, 63, 68]. This allows us to recover sparsity patterns that can be expressed as lying in a union of groups. If a group is selected, then all the coefficients in the group are selected.

- If the groups $G \in \mathcal{G}$ are non overlapping, then (2.17) reduces to the sparse group lasso [76]. Of course, for non overlapping groups, if $\mu = 0$, then we get the standard group lasso [91].

Figure 10 shows the effect that the parameter μ has on the shape of the "ball" $\|\boldsymbol{w}_G\| + \mu\|\boldsymbol{w}_G\|_1 \leq \delta$, for a single two dimensional group G.



(a) $\mu = 0$ (b) $\mu = 0.2$ (c) $\mu = 1$ (d) $\mu = 10$

Figure 10: Effect of μ on the shape of the set $\|\boldsymbol{w}_G\| + \mu\|\boldsymbol{w}_G\|_1 \leq \delta$, for a two dimensional group G. $\mu = 0$ (a) yields the ℓ_2 norm ball. As the value of μ in increased, the effect of the ℓ_1 norm term increases (b) (c). Finally as μ gets very large, the set resembles the ℓ_1 ball (d).

Properties of SOSlasso Penalty

The example in Table 1 gives an insight into the kind of sparsity patterns preferred by the function $h(\boldsymbol{x})$. We will tend to prefer solutions that have a small value of $h(\cdot)$. Consider 3 instances of $\boldsymbol{x} \in \mathbb{R}^{10}$, and the corresponding group lasso, ℓ_1 norm, and $h(\boldsymbol{x})$ function values. The vector is assumed to be made up of two groups, $G_1 = \{1, 2, 3, 4, 5\}$ and $G_2 = \{6, 7, 8, 9, 10\}$. $h(\boldsymbol{x})$ is smallest when the support set is sparse within groups, and also when only one of the two groups is selected (column 5). The ℓ_1 norm does not take into account sparsity across groups (column 4), while the group lasso norm does not take into account sparsity within groups (column 3). Since the groups do not overlap, the latent group lasso penalty reduces to the group lasso penalty and $h(\boldsymbol{x})$ reduces to the sparse group lasso penalty.

Support	**Values**	$\sum_G \|\boldsymbol{x}_G\|$	$\|\boldsymbol{x}\|_1$	$\sum_G (\|\boldsymbol{x}_G\| + \|\boldsymbol{x}_G\|_1)$
$\{1, 4, 9\}$	$\{3, 4, 7\}$	12	14	26
$\{1, 2, 3, 4, 5\}$	$\{2, 5, 2, 4, 5\}$	8.602	18	26.602
$\{1, 3, 4\}$	$\{3, 4, 7\}$	8.602	14	22.602

Table 1: Different instances of a 10-d vector and their corresponding norms.

The next table shows that $h(\boldsymbol{x})$ indeed favors solutions that are not only group sparse, but also exhibit sparsity within groups when the groups overlap. Consider again a 10-dimensional vector $\boldsymbol{x}$ with three overlapping groups $\{1, 2, 3, 4\}$, $\{3, 4, 5, 6, 7\}$ and $\{7, 8, 9, 10\}$. Suppose the vector $\boldsymbol{x} = [0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0]^T$. From the form of the function in (2.17), we see that the vector can be seen as a sum of three vectors $\boldsymbol{w}_i,\ \ i = 1, 2, 3$, corresponding to the three groups listed above. Consider the following instances of the $\boldsymbol{w}_i$ vectors, which are all feasible solutions for the optimization problem in (2.17):

1. $\boldsymbol{w}_1 = [0\ \ 0\ \ -1\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T$, $\boldsymbol{w}_2 = [0\ \ 0\ \ 1\ \ 1\ \ 1\ \ 0\ \ 1\ \ 0\ \ 0\ \ 0]^T$, $\boldsymbol{w}_3 = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^T$

2. $\boldsymbol{w}_1 = [0\ \ 0\ \ 1\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T$, $\boldsymbol{w}_2 = [0\ \ 0\ \ 0\ \ 0\ \ 1\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T$, $\boldsymbol{w}_3 = [0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0]^T$

3. $\boldsymbol{w}_1 = [0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T$, $\boldsymbol{w}_2 = [0\ \ 0\ \ 1\ \ 0\ \ 1\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T$, $\boldsymbol{w}_3 = [0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0]^T$

4. $\boldsymbol{w}_1 = [0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T$, $\boldsymbol{w}_2 = [0\ \ 0\ \ 1\ \ 0\ \ 1\ \ 0\ \ 1\ \ 0\ \ 0\ \ 0]^T$, $\boldsymbol{w}_3 = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^T$

In the above list, the first instance corresponds to the case where the support is localized to two groups, and one of these groups (group 2) has only one zero. The second case corresponds to the case where all 3 groups have non zeros in them. The third case has support localized

to two groups, and both groups are sparse. Finally, the fourth case has only the second group having non zero coefficients, and this group is also sparse. Table 2 shows that the smallest value of the sum of the terms is achieved by the fourth decomposition, and hence that will correspond to the optimal representation.

$A = \|\boldsymbol{w}_1\| + \mu\|\boldsymbol{w}_1\|_1$	$B = \|\boldsymbol{w}_2\| + \mu\|\boldsymbol{w}_2\|_1$	$C = \|\boldsymbol{w}_3\| + \mu\|\boldsymbol{w}_3\|_1$	$A + B + C$
$\sqrt{B} + \mu$	$2\sqrt{B} + 4\mu$	0	$3\sqrt{B} + 5\mu$
$\sqrt{B} + \mu$	$\sqrt{B} + \mu$	$\sqrt{B} + \mu$	$3\sqrt{B} + 3\mu$
0	$\sqrt{2}\sqrt{B} + 2\mu$	$\sqrt{B} + \mu$	$(1 + \sqrt{2})\sqrt{B} + 3\mu$
0	$\sqrt{3}\sqrt{B} + 3\mu$	0	$\sqrt{3}\sqrt{B} + 3\mu$

Table 2: Values of the sum of the ℓ_1 and ℓ_2 norms corresponding to the decompositions listed above. Note that the optimal representation corresponds to the case $\boldsymbol{w}_1 = \boldsymbol{w}_3 = \mathbf{0}$, and $\boldsymbol{w}_2$ being a sparse vector.

Lastly, we can show that $h(\boldsymbol{x})$ is a norm. This will allow us to derive consistency results for the optimization problems we are interested in in this section.

Lemma 2.3.2. *The function*

$$h(\boldsymbol{x}) = \inf_{\{\boldsymbol{w}_G\} \in \mathcal{W}(\boldsymbol{x})} \sum_{G \in \mathcal{G}} \left(\sqrt{B} \|\boldsymbol{w}_G\|_2 + \mu \|\boldsymbol{w}_G\|_1 \right)$$

is a norm

Proof. It is trivial to show that $h(\boldsymbol{x}) \geq 0$ with equality *iff* $\boldsymbol{x} = 0$. We now show positive homogeneity. Suppose $\{\boldsymbol{w}_G\} \in \mathcal{W}(\boldsymbol{x})$ is an optimal representation (Definition 2.3.4) of $\boldsymbol{x}$, and let $\gamma \in \mathbb{R} \backslash \{\mathbf{0}\}$. Then, $\sum_{G \in \mathcal{G}} \boldsymbol{w}_G = \boldsymbol{x} \;\; \Rightarrow \sum_{G \in \mathcal{G}} \gamma \boldsymbol{w}_G = \gamma \boldsymbol{x}$. This leads to the following

set of inequalities:

$$h(\boldsymbol{x}) = \sum_{G\in\mathcal{G}} \left(\sqrt{B}\|\boldsymbol{w}_G\|_2 + \mu\|\boldsymbol{w}_G\|_1\right) = \frac{1}{|\gamma|}\sum_{G\in\mathcal{G}} \left(\sqrt{B}\|\gamma\boldsymbol{w}_G\|_2 + \mu\|\gamma\boldsymbol{w}_G\|_1\right) \geq \frac{1}{|\gamma|}h(\gamma\boldsymbol{x}) \tag{2.19}$$

Now, assuming $\{\boldsymbol{v}_G\} \in \mathcal{W}(\gamma\boldsymbol{x})$ is an optimal representation of $\gamma\boldsymbol{x}$, we have that $\sum_{G\in\mathcal{G}} \frac{\boldsymbol{v}_G}{\gamma} = \boldsymbol{x}$, and we get

$$h(\gamma\boldsymbol{x}) = \sum_{G\in\mathcal{G}} \left(\sqrt{B}\|\boldsymbol{v}_G\|_2 + \mu\|\boldsymbol{v}_G\|_1\right) = |\gamma|\sum_{G\in\mathcal{G}} \left(\sqrt{B}\left\|\frac{\boldsymbol{v}_G}{\gamma}\right\|_2 + \mu\left\|\frac{\boldsymbol{v}_G}{\gamma}\right\|_1\right) \geq |\gamma|h(\boldsymbol{x}) \tag{2.20}$$

Positive homogeneity follows from (2.19) and (2.20). The inequalities are a result of the possibility of the vectors not corresponding to the respective optimal representations.

For the triangle inequality, again let $\{\boldsymbol{w}_G\} \in \mathcal{W}(\boldsymbol{x}), \{\boldsymbol{v}_G\} \in \mathcal{W}(\boldsymbol{y})$ correspond to the optimal representation for $\boldsymbol{x}, \boldsymbol{y}$ respectively. Then by definition,

$$\begin{aligned} h(\boldsymbol{x}+\boldsymbol{y}) &\leq \sum_{G\in\mathcal{G}} (\sqrt{B}\|\boldsymbol{w}_G + \boldsymbol{v}_G\|_2 + \mu\|\boldsymbol{w}_G + \boldsymbol{v}_G\|_1) \\ &\leq \sum_{G\in\mathcal{G}} (\sqrt{B}\|\boldsymbol{w}_G\|_2 + \sqrt{B}\|\boldsymbol{v}_G\|_2 + \mu\|\boldsymbol{w}_G\|_1 + \mu\|\boldsymbol{v}_G\|_1) \\ &= h(\boldsymbol{x}) + h(\boldsymbol{y}) \end{aligned}$$

The first and second inequalities follow by definition and the triangle inequality respectively.

□

2.3.6 Sample Complexity Bounds for the Sparse Overlapping Sets Lasso

From Theorem 2.3.1, we see that sample complexity bounds can be derived for the SOSlasso, if we first devise a constraint set $\mathcal{C}$ for our problem, and then bound the square of the mean width of that set.

The first thing to note, is that we can obtain a bound on the $h(\boldsymbol{x})$ in terms of $\|\boldsymbol{x}\|$. Note that, given a vector $\boldsymbol{x}$, we can always find a representation $\boldsymbol{x} = \sum_{G \in \mathcal{G}} \boldsymbol{u}_G$ such that the supports of $\boldsymbol{u}_G$ do not overlap. Then

$$\begin{aligned} h(\boldsymbol{x}) &\leq \sum_{G \in \mathcal{G}} \sqrt{B}\|\boldsymbol{u}_G\| + \|\boldsymbol{u}_G\|_1 \\ &\leq (\sqrt{B} + \sqrt{\alpha B}) \sum_{G \in \mathcal{G}} \|\boldsymbol{u}_G\| \\ &= \sqrt{kB}(1 + \sqrt{\alpha})\|\boldsymbol{x}\| \end{aligned} \tag{2.21}$$

Since $h(\cdot)$ is a norm, it is convex. This fact, along with (2.21) allows us to define a convex set:

$$\mathcal{C}_{sos} = \left\{ \boldsymbol{x} : \|\boldsymbol{x}\| \leq 1, \;\; h(\boldsymbol{x}) \leq \sqrt{kB}(1 + \sqrt{\alpha}) \right\} \tag{2.22}$$

The SOSlasso optimization problem can then be written as

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} -\boldsymbol{y}^T \boldsymbol{\Phi} \boldsymbol{x} \;\; \textbf{s.t.} \;\; \boldsymbol{x} \in \mathcal{C}_{sos} \tag{2.23}$$

The next Theorem gives a bound on the mean width of the set $\mathcal{C}_{sos}$ above. We state the theorem here and prove the result in Appendix A.2

Theorem 2.3.3. *The mean width of the set*

$$\mathcal{C}_{sos} = \left\{ \boldsymbol{x} : \|\boldsymbol{x}\| \leq 1, \;\; h(\boldsymbol{x}) \leq \sqrt{kB}(1+\sqrt{\alpha}) \right\}$$

can be bounded as

$$\omega(\mathcal{C}_{sos})^2 \leq k\left(1+\sqrt{\alpha}\right)^2 (\sqrt{2\log(M)} + \sqrt{B})^2$$

Armed with the result above, we now obtain the following result:

Theorem 2.3.4. *Suppose there exists a coefficient vector $\boldsymbol{x}^\star$ that is (k,α)-group sparse. Suppose the data matrix $\boldsymbol{\Phi} \in \mathbb{R}^{n\times p}$ and observation model follow the setting in Theorem 2.3.1. Suppose we solve (2.16) for the constraint set given by (2.22). For $\delta > 0$ and a constant C, if the number of measurements satisfies*

$$n \geq C\delta^{-2}k(1+\alpha)\left[\log\left(M\right) + B\right]$$

then the solution of the logistic SOSlasso satisfies

$$\left\| \widehat{\boldsymbol{x}} - \frac{\boldsymbol{x}^\star}{\|\boldsymbol{x}^\star\|_2} \right\|_2^2 \;\leq\; \delta \max(\|\boldsymbol{x}^\star\|^{-1}, 1)$$

Remarks

Theorem 2.3.4 generalizes existing results on sparse logistic regression, and also yields new results for (overlapping) group sparse logistic regression. In particular, we make the following observations:

- When the groups are singletons, we have $B = 1$, and we obtain known results for ℓ_1 penalized regression. Setting $\alpha = 1, \;\; k = s, \;\; M = n$, we get

$$n \geq Cs \log(p)$$

- For traditional group lasso, we merely set $\alpha = 1$ and obtain

$$n \geq Ck(\log(M) + B)$$

- We also obtain results for the sparse group lasso, since it is a special case of the SOSlasso.

In chapter 4, we apply the SOSlasso for classification to problems in fMRI and computational biology.

2.3.7 Extensions to Data with Correlated Entries

The results we proved above can be extended to data $\mathbf{\Phi}$ with correlated Gaussian entries as well (see [69] for results in linear regression settings). Indeed, in most practical applications we are interested in, the features are expected to contain correlations. For example, in the fMRI application that is one of the major motivating applications of our work, it is reasonable to assume that voxels in the brain will exhibit correlation amongst themselves at a given time instant. This entails scaling the number of measurements by the condition number of the covariance matrix $\mathbf{\Sigma}$, where we assume that each row if the measurement matrix $\mathbf{\Phi}$ is sampled from a Gaussian $(0, \mathbf{\Sigma})$ distribution. Specifically, we obtain the following generalization of the result in [65] for the SOSlasso with a correlated Gaussian design.

We now consider the following constraint set:

$$\mathcal{C}_{corr} = \{\boldsymbol{x} : h(\boldsymbol{x}) \leq \frac{1}{\sigma_{min}(\boldsymbol{\Sigma}^{\frac{1}{2}})}\sqrt{kB}(1+\sqrt{\alpha}), \|\boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{x}\| \leq 1\} \tag{2.24}$$

We consider the set $\mathcal{C}_{corr}$ and not $\mathcal{C}_{sos}$ in (2.22), since we require the constraint set to be a subset of the unit Euclidean ball. In the proof of Corollary 2.3.5 below, we will reduce the problem to an optimization over variables of the form $\boldsymbol{z} = \boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{x}$, and hence we require $\|\boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{x}\|_2 \leq 1$. Enforcing this constraint leads to the corresponding upper bound on $h(\boldsymbol{x})$.

Corollary 2.3.5. *Let the entries of the data matrix $\boldsymbol{\Phi}$ be sampled from a $\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ distribution. Suppose the measurements follow the model in (2.15). Suppose we wish to recover a $(k, \alpha)-$ group sparse vector from the set $\mathcal{C}_{corr}$ in (2.24). Suppose the true coefficient vector $\boldsymbol{x}^\star$ satisfies $\|\boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{x}^\star\| = 1$. Then, so long as the number of measurements n satisfies*

$$n \geq C\delta^{-2}k(1+\alpha)(\sqrt{2\log(M)} + \sqrt{B})^2\kappa(\boldsymbol{\Sigma})$$

the solution to (2.16) satisfies

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}^\star\|^2 \leq \frac{\delta}{\sigma_{min}(\boldsymbol{\Sigma})}$$

where $\sigma_{min}(\cdot)$, $\sigma_{max}(\cdot)$ and $\kappa(\cdot)$ denote the minimum and maximum singular values and the condition number of the corresponding matrices respectively.

We prove this result in Appendix A.3

2.3.8 The SOSlasso for Linear Regression

Up until now, we have been focussing on classification settings, under a logistic regression model. In this section, we derive consistency bounds for the SOSlasso under linear regression settings. We consider the following optimization problem:

$$\widehat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \left\{ \frac{1}{2n} \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}\|_2^2 + \lambda_n h(\boldsymbol{x}) \right\} \tag{2.25}$$

In the sequel, for brevity, we define $\mathcal{L}_{\boldsymbol{\Phi}}(\boldsymbol{x}) := \frac{1}{2n}\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}\|_2^2$.

In the previous sections, we showed that $h(\boldsymbol{x})$ is a norm. The dual norm of $h(\boldsymbol{x})$ can be bounded as

$$\begin{aligned}
h^*(\boldsymbol{u}) &= \max_{\boldsymbol{x}} \{\boldsymbol{x}^T\boldsymbol{u}\} \;\; \textbf{s.t.} \;\; h(\boldsymbol{x}) \leq 1 \\
&= \max_{\mathcal{W}} \{\sum_{G\in\mathcal{G}} \boldsymbol{w}_G^T \boldsymbol{u}_G\} \;\; \textbf{s.t.} \;\; \sum_{G\in\mathcal{G}} (\|\boldsymbol{w}_G\|_2 + \|\boldsymbol{w}_G\|_1) \leq 1 \\
&\overset{(i)}{\leq} \max_{\mathcal{W}} \{\sum_{G\in\mathcal{G}} \boldsymbol{w}_G^T \boldsymbol{u}_G\} \;\; \textbf{s.t.} \;\; \sum_{G\in\mathcal{G}} 2\|\boldsymbol{w}_G\|_2 \leq 1 \\
&= \max_{\mathcal{W}} \{\sum_{G\in\mathcal{G}} \boldsymbol{w}_G^T \boldsymbol{u}_G\} \;\; \textbf{s.t.} \;\; \sum_{G\in\mathcal{G}} \|\boldsymbol{w}_G\|_2 \leq \frac{1}{2} \\
\Rightarrow h^*(\boldsymbol{u}) &\leq \max_{G\in\mathcal{G}} \frac{1}{2}\|\boldsymbol{u}_G\|_2
\end{aligned} \tag{2.26}$$

(i) follows from the fact that the constraint set in (i) is a superset of the constraint set in the previous statement, since $\|\boldsymbol{a}\|_2 \leq \|\boldsymbol{a}\|_1$. (2.26) follows from noting that the maximum is obtained by setting $\boldsymbol{w}_{G^*} = \frac{\boldsymbol{u}_{G^*}}{2\|\boldsymbol{u}_{G^*}\|_2}$, where $G^* = \arg\max_{G\in\mathcal{G}} \|\boldsymbol{u}_G\|_2$. The inequality (2.26) is far more tractable than the actual dual norm, and will be useful in our derivations below. Since $h(\cdot)$ is a norm, we can apply methods developed in [59] to derive consistency rates for

the optimization problem (2.25).

Definition A norm $h(\cdot)$ is decomposable with respect to the subspace pair $sA \subset sB$ if $h(\boldsymbol{a} + \boldsymbol{b}) = h(\boldsymbol{a}) + h(\boldsymbol{b}) \;\; \forall \boldsymbol{a} \in sA, \boldsymbol{b} \in sB^{\perp}$.

Lemma 2.3.6. *Let $\boldsymbol{x}^{\star} \in \mathbb{R}^p$ be a vector that can be decomposed into (overlapping) groups with within-group sparsity. Let $\mathcal{G}^{\star} \subset \mathcal{G}$ be the set of active groups of $\boldsymbol{x}^{\star}$. Let $S = supp(\boldsymbol{x}^{\star})$ indicate the support set of $\boldsymbol{x}$. Let sA be the subspace spanned by the coordinates indexed by S, and let $sB = sA$. We then have that the norm in (2.17) is decomposable with respect to sA, sB*

The result follows in a straightforward way from noting that supports of decompositions for vectors in sA and $sB^{\perp}$ do not overlap.

Proof. Let $\boldsymbol{a} \in sA$ and $\boldsymbol{b} \in sB^{\perp}$ be two vectors. Let $\boldsymbol{w}^A$ and $\boldsymbol{w}^B$ correspond to the vectors in the optimal decompositions of $\boldsymbol{a}$ and $\boldsymbol{b}$ respectively. Note that $S \subset \bigcup_{G \in \mathcal{G}^{\star}} G$. Since the vectors $\boldsymbol{w}^A$ and $\boldsymbol{w}^B$ are the optimal decompositions, we have that none of the supports of the vectors $\boldsymbol{w}^A$ overlap with those in $\boldsymbol{w}^B$. Hence,

$$\begin{aligned} h(\boldsymbol{a}) + h(\boldsymbol{b}) &= \sum_{G \in \mathcal{G}^{\star}} \left(\|\boldsymbol{w}_G^A\| + \|\boldsymbol{w}_G^A\|_1 \right) + \sum_{G \in \mathcal{G}} \left(\|\boldsymbol{w}_G^B\| + \|\boldsymbol{w}_G^B\|_1 \right) \\ &= \sum_{G \in \mathcal{G}} \left(\|\boldsymbol{w}_G^A\| + \|\boldsymbol{w}_G^B\| + \|\boldsymbol{w}_G^A\|_1 + \|\boldsymbol{w}_G^B\|_1 \right) = h(\boldsymbol{a} + \boldsymbol{b}) \end{aligned}$$

□

Definition Given a subspace sB, the subspace compatibility constant with respect to a norm $\| \; \|$ is given by

$$\Psi(B) = \sup \left\{ \frac{h(\boldsymbol{x})}{\|\boldsymbol{x}\|} \;\; \forall \boldsymbol{x} \in sB \backslash \{\boldsymbol{0}\} \right\}$$

Lemma 2.3.7. *Consider a vector $\boldsymbol{x}$ that can be decomposed into $\mathcal{G}^\star \subset \mathcal{G}$ active groups. Suppose the maximum group size is B, and also assume that a fraction $\alpha \in (0,1)$ of the coordinates in each active group is non zero. Then,*

$$h(\boldsymbol{x}) \leq (1+\sqrt{B\alpha})\sqrt{|\mathcal{G}^\star|}\|\boldsymbol{x}\|_2$$

Proof. For any vector $\boldsymbol{x}$ with $supp(\boldsymbol{x}) \subset \mathcal{G}^\star$, there exists a representation $\boldsymbol{x} = \sum_{G\in\mathcal{G}^\star} \boldsymbol{w}_G$, such that the supports of the different $\boldsymbol{w}_G$ do not overlap. Then,

$$h(\boldsymbol{x}) \leq \sum_{G\in\mathcal{G}^\star} (\|\boldsymbol{w}_G\|_2 + \|\boldsymbol{w}_G\|_1) \;\; \leq (1+\sqrt{B\alpha}) \sum_{G\in\mathcal{G}^\star} \|\boldsymbol{w}_G\|_2 \;\; \leq (1+\sqrt{B\alpha})\sqrt{|\mathcal{G}^\star|}\|\boldsymbol{x}\|_2$$

□

We see that $(1+\sqrt{B\alpha})\sqrt{|\mathcal{G}^\star|}$ (Lemma 2.3.7) gives an upper bound on the subspace compatibility constant with respect to the ℓ_2 norm for the subspace indexed by the support of the vector, which is contained in the span of the union of groups in $\mathcal{G}^\star$.

Definition For a given set S, and given vector $\boldsymbol{x}^\star$, the loss function $\mathcal{L}_{\boldsymbol{\Phi}}(\boldsymbol{x})$ satisfies the Restricted Strong Convexity(RSC) condition with parameter κ and tolerance τ if

$$\mathcal{L}_{\boldsymbol{\Phi}}(\boldsymbol{x}^\star+\Delta) - \mathcal{L}_{\boldsymbol{\Phi}}(\boldsymbol{x}^\star) - \langle \nabla\mathcal{L}_{\boldsymbol{\Phi}}(\boldsymbol{x}^\star), \Delta\rangle \geq \kappa\|\Delta\|_2^2 - \tau^2(\boldsymbol{x}^\star) \;\; \forall \Delta \in S$$

In this paper, we consider vectors $\boldsymbol{x}^\star$ that lie *exactly* in $k \ll M$ groups, and display within-group sparsity. This implies that the tolerance $\tau(\boldsymbol{x}^\star) = 0$, and we will ignore this term henceforth.

We also define the following set, which will be used in the sequel:

$$C(sA, sB, \boldsymbol{x}^\star) := \{\Delta \in \mathbb{R}^p | h(\Pi_{sB^\perp}\Delta) \leq 3h(\Pi_{sB}\Delta) + 4h(\Pi_{sA^\perp}\boldsymbol{x}^\star)\} \tag{2.27}$$

where $\Pi_{sA}(\cdot)$ denotes the projection onto the subspace sA. Based on the results above, we can now apply a result from [59] to the SOSlasso:

Theorem 2.3.8. *(Corollary 1 in [59]) Consider a convex and differentiable loss function such that RSC holds with constants κ and $\tau = 0$ over (2.27), and a norm $h(\cdot)$ decomposable over sets sA and sB. For the optimization program in (2.25), using the parameter $\lambda_n \geq 2h^*(\nabla\mathcal{L}_{\boldsymbol{\Phi}}(\boldsymbol{x}^\star))$, any optimal solution $\hat{\boldsymbol{x}}_{\lambda_n}$ to (2.25) satisfies*

$$\|\widehat{\boldsymbol{x}}_{\lambda_n} - \boldsymbol{x}^\star\|_2^2 \leq \frac{9\lambda_n^2}{\kappa}\Psi^2(sB)$$

The result above shows a general bound on the error using the lasso with sparse overlapping sets. Note that the regularization parameter λ_n as well as the RSC constant κ depend on the loss function $\mathcal{L}_{\boldsymbol{\Phi}}(\boldsymbol{x})$. In the next section, we consider the least squares loss (2.25), and show that the estimate using the SOSlasso is consistent.

2.3.9 Consistency of SOSlasso with Squared Error Loss

We first need to bound the dual norm of the gradient of the loss function, so as to bound λ_n. Consider $\mathcal{L} := \mathcal{L}_{\boldsymbol{\Phi}}(\boldsymbol{x}) = \frac{1}{2n}\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}\|^2$. The gradient of the loss function with respect to $\boldsymbol{x}$ is given by $\nabla\mathcal{L} = \frac{1}{n}\boldsymbol{\Phi}^T(\boldsymbol{\Phi}\boldsymbol{x} - \boldsymbol{y}) = \frac{1}{n}\boldsymbol{\Phi}^T\eta$

where $\eta = [\eta_1^T\eta_2^T \ldots \eta_{\mathcal{T}}^T]^T$ (see Section 2.2.1). Our goal now is to find an upper bound on

the quantity $h^*(\nabla\mathcal{L})$, which from (2.26) is

$$\frac{1}{2}\max_{G\in\mathcal{G}}\|\nabla\mathcal{L}_G\|_2 = \frac{1}{2n}\max_{G\in\mathcal{G}}\|\boldsymbol{\Phi}_G^T\eta\|_2$$

where $\boldsymbol{\Phi}_G$ is the matrix $\boldsymbol{\Phi}$ restricted to the columns indexed by the group G. We will prove an upper bound for the above quantity in the course of the results that follow.

Since $\eta \sim \mathcal{N}(0, \sigma^2\boldsymbol{I})$, we have $\boldsymbol{\Phi}_G^T\eta \sim \sigma\mathcal{N}(0, \boldsymbol{\Phi}_G^T\boldsymbol{\Phi}_G)$. Defining $\sigma_{mG} := \sigma_{\max}\{\boldsymbol{\Phi}_G^T\boldsymbol{\Phi}_G\}$ to be the maximum singular value, we have $\|\boldsymbol{\Phi}_G^T\eta\|_2^2 \leq \sigma^2\sigma_{mG}^2\|\gamma\|_2^2$, where $\gamma \sim \mathcal{N}(0, \boldsymbol{I}_{|G|}) \Rightarrow \|\gamma\|_2^2 \sim \chi^2_{|G|}$, where χ^2_d is a chi-squared random variable with d degrees of freedom. This allows us to work with the more tractable chi squared random variable when we look to bound the dual norm of $\nabla\mathcal{L}$. The next lemma helps us obtain a bound on the maximum of χ^2 random variables.

Lemma 2.3.9. *Let $\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_M$ be chi-squared random variables with d degrees of freedom. Then for some constant c,*

$$\mathbb{P}\left(\max_{i=1,2,\ldots,M} z_i \leq c^2 d\right) \geq 1 - \exp\left(\log(M) - \frac{(c-1)^2 d}{2}\right)$$

Proof. From the chi-squared tail bound in [22], $\mathbb{P}(z_i \geq c^2 d) \leq \exp\left(-\frac{(c-1)^2 d}{2}\right)$. The result follows from a union bound and inverting the expression. □

Lemma 2.3.10. *Consider the loss function $\mathcal{L} := \frac{1}{2n}\sum_{t=1}^{\mathcal{T}}\|\boldsymbol{y}_t - \boldsymbol{\Phi}_t\boldsymbol{x}_t\|^2 = \frac{1}{2n}\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}\|^2$, with the $\boldsymbol{\Phi}_t's$ deterministic and the measurements corrupted with AWGN of variance σ^2. For the regularizer in (2.17), the dual norm of the gradient of the loss function is bounded as*

$$h^*(\nabla\mathcal{L})^2 \leq \frac{\sigma^2\sigma_m^2}{4}\frac{(\log(M) + B)}{n}$$

with probability at least $1 - c_1 \exp(-c_2 n)$*, for* $c_1, c_2 > 0$*, and where* $\sigma_m = \max_{G \in \mathcal{G}} \sigma_{mG}$

Proof. Let $\gamma \sim \chi^2_{|G|}$. We begin with the upper bound obtained for the dual norm of the regularizer in (2.26):

$$\begin{aligned} h^*(\nabla \mathcal{L})^2 &\overset{(i)}{\leq} \frac{1}{4} \max_{G \in G} \left\| \frac{1}{n} \mathbf{\Phi}_G^T \eta \right\|_2^2 \\ &\leq \frac{\sigma^2}{4} \max_{G \in \mathcal{G}} \frac{\sigma_{mG}^2 \gamma}{n^2} \\ &\overset{(ii)}{\leq} \frac{\sigma^2 \sigma_m^2}{4} \max_{G \in \mathcal{G}} \frac{\gamma}{n^2} \\ &\overset{(iii)}{\leq} \frac{\sigma^2 \sigma_m^2}{4} c^2 B \quad \textbf{w. p. } 1 - \exp\left(\log(M) - \frac{(cn-1)^2 B}{2} \right) \end{aligned}$$

where (i) follows from the formulation of the gradient of the loss function and the fact that the square of maximum of non negative numbers is the maximum of the squares of the same numbers. In (ii), we have defined $\sigma_m = \max_G \sigma_{mG}$. Finally, we have made use of Lemma 2.3.9 in (iii). We then set

$$c^2 = \frac{\log(M) + B}{Bn}$$

to obtain the result. □

We combine the results developed so far to derive the following consistency result for the SOS lasso, with the least squares loss function.

Theorem 2.3.11. *Suppose we obtain linear measurements of a sparse overlapping grouped matrix* $\boldsymbol{X}^\star \in \mathbb{R}^{p \times \mathcal{T}}$*, corrupted by AWGN of variance* σ^2*. Suppose the matrix* $\boldsymbol{X}^\star$ *can be decomposed into* M *possible overlapping groups of maximum size* B*, out of which* k *are active. Furthermore, assume that a fraction* $\alpha \in (0, 1]$ *of the coefficients are non zero in each*

active group. Consider the following SOSlasso regression problem:

$$\widehat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \left\{ \frac{1}{2n} \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}\|_2^2 \;\; + \;\; \lambda_n h(\boldsymbol{x}) \right\},$$

$$h(\boldsymbol{x}) = \inf_{\mathcal{W}} \sum_{G \in \mathcal{G}} (\|\boldsymbol{w}_G\|_2 + \|\boldsymbol{w}_G\|_1) \quad \textbf{\textit{s.t.}} \sum_{G \in \mathcal{G}} \boldsymbol{w}_G = \boldsymbol{x}$$

Suppose the data matrices $\boldsymbol{\Phi}_t$ are non random, and the loss function satisfies restricted strong convexity assumptions with parameter κ. Then, for $\lambda_n^2 \geq \frac{\sigma^2 \sigma_m^2 (\log(M)+B)}{4n}$, the following holds with probability at least $1 - c_1 \exp(-c_2 n)$, with $c_1, c_2 > 0$:

$$\|\widehat{\boldsymbol{x}} - \boldsymbol{x}^\star\|_2^2 \leq \frac{9}{4} \frac{\sigma^2 \sigma_m^2 \left(1 + \sqrt{B\alpha}\right)^2 k(\log(M) + B)}{n\kappa}$$

where we define $\sigma_m := \max_{G \in \mathcal{G}} \sigma_{max}\{\boldsymbol{\Phi}_G^T \boldsymbol{\Phi}_G\}$

Proof. Follows from substituting in Theorem 2.3.8 the results from Lemma 2.3.7 and Lemma 2.3.10. □

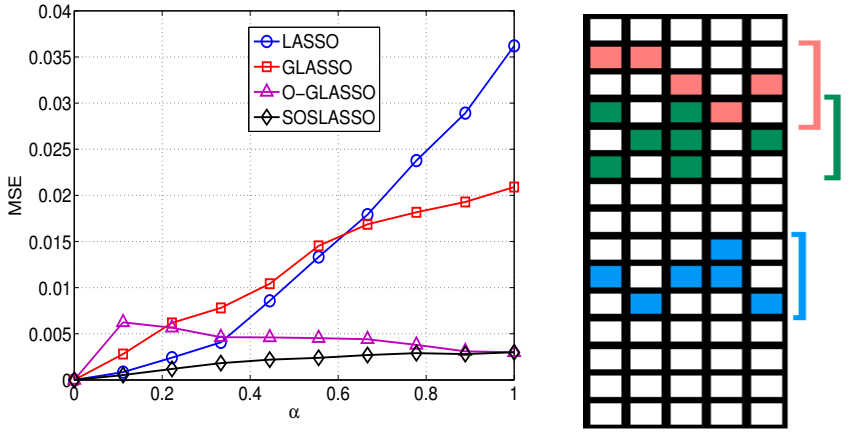
2.3.10 Experiments : Toy Data, Linear Regression

We show that the method we propose can also be applied to the linear regression setting. To this end, we consider simulated data and a multitask linear regression setting, and look to recover the coefficient matrix. We also use the simulated data to study the properties of the function we propose in (2.17).

The toy data is generated as follows: we consider $\mathcal{T} = 20$ tasks, and consider overlapping groups of size $B = 6$. The groups are defined so that neighboring groups overlap ($G_1 = \{1, 2, \ldots, 6\}$, $G_2 = \{5, 6, \ldots, 10\}$, $G_3 = \{9, 10, \ldots, 14\}$, $\ldots$). We consider a case with

$M = 100$ groups, We set $k = 10$ groups to be active. We vary the sparsity level of the active groups α and obtain $m = 100$ Gaussian linear measurements corrupted with Additive White Gaussian Noise of standard deviation $\sigma = 0.1$. We repeat this procedure 100 times and average the results. To generate the coefficient matrices $\boldsymbol{X}^\star$, we select k groups at random, and within the active groups, only retain fraction α of the coefficients, again at random. The retained locations are then populated with uniform $[-1, 1]$ random variables.

The regularization parameters were clairvoyantly picked to minimize the Mean Squared Error (MSE) over a range of parameter values. The results of applying lasso, standard latent group lasso [37], Group lasso where each group corresponds to a row of the sparse matrix, [49] and our SOSlasso to these data are plotted in Figures 11a, varying α.



(a) Varying α (b) Sample pattern

Figure 11: Figure (a) shows the result of varying α. The SOSlasso accounts for both inter and intra group sparsity, and hence performs the best. The Glasso achieves good performance only when the active groups are non sparse. Figure (b) shows a toy sparsity pattern, with different colors and brackets denoting different overlapping groups

Figure 11a shows that, as the sparsity within the active group reduces (i.e. the active groups become more dense), the overlapping group lasso performs progressively better. This

is because the overlapping group lasso does not account for sparsity within groups, and hence the resulting solutions are far from the true solutions for small values of α. The SOSlasso however does take this into account, and hence has a lower error when the active groups are sparse. Note that as $\alpha \to 1$, the SOSlasso approaches O-Glasso [37]. The Lasso [83] does not account for group structure at all and performs poorly when α is large, whereas the Group lasso [49] does not account for overlapping groups, and hence performs worse than O-Glasso and SOSlasso.

2.4 Conclusions

In this chapter, we studied theoretical properties of structured pattern recovery. We analyzed the group lasso with overlapping groups, and showed that the number of (sub)Gaussian measurements needed for recovery does not depend on the extent of overlap between groups, but only on the number of non zero groups and the total number of groups.

We then introduced the Sparse Overlapping Sets lasso, a framework for structured pattern recovery when one is looking for patterns that are sparse across groups, as well as sparse within groups. We again derived sample complexity bounds for the convex SOSlasso problem under both classification and regression settings. The SOSlasso can be specialized to both the lasso and the (overlapping) group lasso, and hence provides a very general framework to perform structured sparse pattern recovery in high dimensional data.

Chapter 3

CoGEnT : A Greedy Framework for Structurally Constrained Signal Recovery

In this chapter, we focus on a general purpose algorithm to solve structurally constrained high dimensional problems. In Chapter 1, we defined the notion of atoms, atomic norms and atomic sets. We showed how many common applications in information processing can be seen as recovery of a signal that has a constraint on it's simplicity. In this chapter, we give further examples of atomic sets, and then pose the following problem:

Can a common algorithmic framework be used to solve atomic norm constrained recovery problems?

We show that the answer to the above question is affirmative, and devise a method to perform atomic norm constrained recovery. We propose a greedy scheme based on the Conditional Gradient method. We build on the conditional gradient method, by incorporating an efficient step to enhance the quality of the solution at each iteration. Furthermore, to alleviate the inherent drawbacks of greedy schemes, we incorporate backward (or truncation) steps in the method to "parsimonize" the solution at each iteration. The resulting method is a scheme that retains the computational advantages of greedy methods, while providing excellent results on a wide variety of applications.

3.1 Introduction

Minimization of a convex loss function with a constraint on the "simplicity" of the solution has found widespread applications in communications, machine learning, image processing, genetics, and other fields. While exact formulations of the simplicity requirement are often intractable, it is sometimes possible to devise tractable formulations via convex relaxation that are (nearly) equivalent. Since these formulations differ so markedly across applications, a principled and unified convex heuristic for different notions of simplicity has been proposed using notions of *atoms* and *atomic norms* [14]. Atoms are fundamental basis elements of the representation of a signal, chosen so that "simplicity" equates to "representable in terms of a small number of atoms." We list several applications, describing for each application a choice of atoms that captures the concept of simplicity for those applications.

For instance, a sparse signal $\boldsymbol{x}$ may be represented as $\boldsymbol{x} = \sum_{j\in\mathcal{S}} c_j e_j$, where the e_j are the standard unit vectors and $\mathcal{S}$ captures the support of x. One can view the set $\{\pm e_j\}$ as *atoms* that constitute the signal, and the convex hull of these atoms is a set of fundamental importance called the *atomic-norm ball*. The operation of inflation/deflation of the atomic norm ball induces a norm (the *atomic norm*), which serves as an effective regularizer (see Sec. 3.1.1 for details). The atomic set $\{\pm e_j,\ j = 1, 2, \ldots, p\}$ induces the ℓ_1 norm [13], now well-known to be an effective regularizer for sparsity. However, this idea can be generalized. For instance, the atomic norm induced by the convex hull of all unit rank matrices is the nuclear norm, often used as a heuristic for rank minimization [12, 70]. Other novel applications of the atomic-norm framework include the following.

- **Group-norm-constrained multitask learning** problems with group-ℓ_2 norms [2, 37, 67] or group-ℓ_∞ norms [50, 60, 85] have as atoms unit Euclidean balls and unit ℓ_∞-norm

balls, respectively, restricted to specific groups of variables.

- **Group lasso with overlapping groups** arises from applications in genomics, image processing, and machine learning [37, 67]. We showed in the previous chapter that the latent group lasso norm is indeed an atomic norm.

- **Moment problems**, which arise in applications such as radar, communications, seismology, and sensor arrays, have an atomic set which is uncountably infinite [81]. Each atom is a trigonometric moment sequence of an atomic measure supported on the unit interval [81]. This methodology can be extended to signal classes such as Bessel functions, Gaussians, and wavelets.

- **Group testing on graphs** and network tomography finds widespread applications in sensor, computer, social, and biological networks [18, 37]. In such applications, it is typically required to identify a set of faulty edges/nodes from measurements that are based on the known structure of the graph. Each atom can be defined as a subset of nodes or edges in the graph.

- **Hierarchical norms** arise in topic modeling [43], climate and oceanology applications [15], and fMRI data analysis [66]. The atoms here are hybrids of group-sparse and sparse atoms.

- **OSCAR**-regularized problems use an octagonal penalty to simultaneously identify a sparse set of pairwise correlated variables [9]. The atoms are vectors containing at most two nonzeros, with each nonzero entry being $\pm\frac{1}{\sqrt{2}}$ and the signed canonical basis vectors, in two dimensions. In higher dimensions, the OSCAR penalty has been shown to be an atomic norm [92]

- **Tensor Completion:** Signals modeled as tensors have recently enjoyed renewed interest in machine learning [1]. In the case we consider here, in which the tensor is symmetric, orthogonally decomposable, and low (symmetric) rank, the atoms consist of unit-rank symmetric tensors.

- **Deconvolution** is the problem of splitting a signal $z = x + y$ into its constituent components x and y [52], where x and y are succinct with respect to different sets of atoms. Typical cases include the atomic sets being sparse and low rank [89], sparse in the canonical and discrete cosine transform (DCT) bases, and sparse and group sparse [40].

We present a general method called CoGEnT (for "Conditional Gradient with Enhancement and Truncation") that can be applied to general atomic norm problems, in particular to all the applications discussed above. CoGEnT reconstructs signals by minimizing a least-squares loss function that measures the difference between the signal representation and the observations, subject to a "simplicity" constraint on the signal, imposed in terms of an atomic norm. Besides its generality, novel aspects of CoGEnT include (a) introduction of *enhancement steps* at each iteration to improve solution fidelity, (b) introduction of efficient *backward steps* that dramatically improves the performance, (c) introduction of the notion of *inexactness* in the forward step. A comprehensive convergence result is presented.

3.1.1 Preliminaries and Notation

We assume the existence of a known atomic set $\mathcal{A}$ and an unknown signal $\boldsymbol{x}$ in some "ambient" space, where $\boldsymbol{x}$ is a superposition of a small number of atoms from $\mathcal{A}$. (We emphasize that the set of atoms need not be finite.) We assume further that the set $\mathcal{A}$ is symmetric about the origin, that is, $\boldsymbol{a} \in \mathcal{A} \Rightarrow -\boldsymbol{a} \in \mathcal{A}$. The representation of $\boldsymbol{x}$ as a conic combination of atoms

$\boldsymbol{a} \in \mathcal{A}_t$ in a subset $\mathcal{A}_t \subset \mathcal{A}$ is written as follows:

$$\boldsymbol{x} = \sum_{\boldsymbol{a} \in \mathcal{A}_t} c_{\boldsymbol{a}} \boldsymbol{a}, \quad \text{with } c_{\boldsymbol{a}} \geq 0 \text{ for all } \boldsymbol{a} \in \mathcal{A}_t. \tag{3.1}$$

where the $c_{\boldsymbol{a}}$ are scalar coefficients. We write

$$\boldsymbol{x} \in \text{co}(\mathcal{A}_t, \tau), \tag{3.2}$$

for some given $\tau \geq 0$, if it is possible to represent the vector $\boldsymbol{x}$ in the form (3.1), with the additional constraint

$$\sum_{\boldsymbol{a} \in \mathcal{A}_t} c_{\boldsymbol{a}} \leq \tau. \tag{3.3}$$

We use $\boldsymbol{A}_t$ to denote a linear operator which maps the coefficient vector c (with cardinality $|\mathcal{A}_t|$) to a vector in the ambient space, using the vectors in $\mathcal{A}_t$, that is,

$$\boldsymbol{A}_t c := \sum_{\boldsymbol{a} \in \mathcal{A}_t} c_{\boldsymbol{a}} \boldsymbol{a}. \tag{3.4}$$

Since there is a one-to-one relationship between $\mathcal{A}_t$ and the linear operator $\boldsymbol{A}_t$, we use the notation (3.4) more often, and sometimes slightly abuse terminology by referring to $\boldsymbol{A}_t$ as the "basis" at iteration t. We sometimes refer to the "columns" of $\boldsymbol{A}_t$, by which we mean the elements of the corresponding basis $\mathcal{A}_t$.

The *atomic norm* [14] is the gauge functional induced by $\mathcal{A}$:

$$\|\boldsymbol{x}\|_{\mathcal{A}} := \inf\{t > 0 : \boldsymbol{x} \in t(\text{conv}(\mathcal{A}))\}, \tag{3.5}$$

where conv $(\cdot)$ denotes the convex hull of a collection of points. Equivalently, we have

$$\|\boldsymbol{x}\|_{\mathcal{A}} := \inf \left\{ \sum_{\boldsymbol{a} \in \mathcal{A}} c_a : \boldsymbol{x} = \sum_{\boldsymbol{a} \in \mathcal{A}} c_a \boldsymbol{a}, \;\; c_a \geq 0 \right\}. \tag{3.6}$$

Given a representation (3.1), the sum of coefficients in (3.3) is an *upper bound* on the atomic norm $\|x\|_{\mathcal{A}}$. The dual atomic norm is given by

$$\|\boldsymbol{x}\|_{\mathcal{A}}^* = \sup_{\|\boldsymbol{u}\|_{\mathcal{A}} \leq 1} \langle \boldsymbol{u}, \boldsymbol{x} \rangle. \tag{3.7}$$

The dual atomic norm is key to our approach — the atom selection step in CoGEnT amounts to choosing the argument that achieves the supremum in (3.7), for a particular choice of $\boldsymbol{x}$.

Our algorithm CoGEnT solves the convex optimization problem:

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) := \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{\Phi x}\|_2^2 \quad \text{s.t.} \quad \|\boldsymbol{x}\|_{\mathcal{A}} \leq \tau, \tag{3.8}$$

where $\boldsymbol{y} = \boldsymbol{\Phi x} + \boldsymbol{w}$ corresponds to observed measurements, with noise vector $\boldsymbol{w}$. The regularizing constraint on the atomic norm of $\boldsymbol{x}$ enforces "simplicity" with respect to the chosen atomic set. Efficient algorithms are known for this problem when the atoms are standard unit vectors $\pm e_j$ (for which the atomic norm is the ℓ_1 norm) [83, 84, 90] and rank-one matrices (for which the atomic norm is the nuclear norm) [37, 41, 91]. CoGEnT targets the general formulation (3.8), opening up a suite of new applications with rigorous convergence guarantees and state-of-the-art empirical performance.

We remark that while (3.8) is a convex formulation, tractable algorithms for solving it are not known in full generality. Indeed, characterization of the atomic norm is itself intractable in some cases. From an optimization perspective, interior point methods are often impractical,

being either difficult to formulate or too slow for large-scale instances. First order greedy methods are often the methods of choice. Greedy schemes are popular in high dimensional signal recovery settings because of their computational efficiency, scalability to large datasets, and interesting global rate-of-convergence properties. They have found widespread use in large scale machine learning applications [20, 27, 29, 32, 38, 58, 82].

3.1.2 Past Work: Conditional Gradient Method

A conditional gradient (CG) algorithm for (3.8) was introduced in [82]. This greedy approach is often known as "Frank-Wolfe" after the authors who proposed it in the 1950s [31]. At each iteration, it finds the atom that optimizes a first-order approximation to the objective over the feasible region, and adds this atom to the basis for the solution. Each iteration of CoGEnT performs a "forward step" of this type. Although CoGEnT includes various enhancements, it is this forward step that drives the convergence theory, which is similar to that of standard conditional gradient methods [29, 82], although with a different treatment of inexactness in the choice of search direction.

Although greedy methods require more iterations than such prox-linear methods as SpaRSA [90], FISTA [7], and Nesterov's accelerated gradient method [61], each iteration is typically less expensive. For example, in matrix completion applications, prox-linear methods require computation of an SVD of a matrix [11] (or at least a substantial part of it), while CG requires only the computation of the leading singular vectors. In other applications, such as structural SVM [47], CG schemes are the only practical way to solve the optimization formulation. Latent group lasso [37] can be extended to perform regression on very large signals by employing a "replication" strategy, but as the amount of group overlap increases, prox-linear

methods quickly become memory intensive. CG offers a scalable method to solve problems of this form. The procedure to choose each new atom has a linear objective, as opposed to the quadratic program required to perform projection steps in prox-linear methods. The linear subproblem is often easier to solve; in some applications, it makes the difference between tractability and intractability. Moreover, the linear problem need only be solved approximately to retain convergence guarantees [32, 38].

3.1.3 Backward (Truncation) Steps

In signal processing applications, one is interested not only in minimizing the loss function, but also in the "simplicity" of the solutions. For example, when the solution corresponds to the wavelet coefficients of an image, sparsity of the representation is key to its usefulness as a compact representation. In this regard, the basic CG and indeed all greedy schemes suffer from a significant drawback: atoms added at some iterations may be superseded by others added at later iterations, and ultimately may not contribute much to reducing the loss function. By the time the loss function has been reduced to an acceptable level, the basis may contain many such atoms of dubious usefulness, thus detracting from the quality of the solution.

Backward steps in CoGEnT allow atoms to be removed from the basis when they are found to be unhelpful in reducing the objective. We define this step in a flexible way, the only requirement being that it does not degrade the objective function too greatly in comparison to the gain that was obtained at the most recent "forward" iteration. The enhancement / reoptimization step discussed below is one way to perform truncation; we can simply discard those atoms whose coefficients are reduced to zero when we reoptimize over the current basis. This step may be expensive to implement, so we seek alternatives. One such alternative is to test

one-by-one the effect of removing each atom in the current basis — an operation that can be performed efficiently because of the least-squares nature of the loss function in (3.8) — and remove the atom(s) that do not deteriorate the objective beyond a specified limit. A third alternative is to seek a completely new set of basis atoms that can be combined to obtain a vector with similar objective value to the latest iteration.

We note that the backward steps in CoGEnT are quite different from the "away steps" analyzed in [34, 38]. These steps move in the opposite to the "worst possible" linearized direction, and thus *add* a new element to the basis at each iteration, rather than *removing* elements, as we do here. While away steps have been shown to improve the convergence properties of CG method, they do not contribute to enhancing sparsity of the solution.

Forward-backward greedy schemes for ℓ_1 constrained minimization have been considered previously in [39,44,48,93]. These methods build on the Orthogonal Matching Pursuit (OMP) algorithm [84], and cannot be readily extended to the general setting (3.8).

3.1.4 Enhancement (Reoptimization) Steps

The enhancement / reoptimization step in CoGEnT takes the current basis and seeks a new set of coefficients in the representation (3.4) that reduces the objective while satisfying the norm constraint. (A "full correction" step of this type was described in [38].) The step is implemented as a linear least-squares objective over a simplex. CoGEnT solves it with a gradient projection method, using a warm start based on the current set of coefficients. Projection onto the simplex can be performed in $O(n_{t+1})$ operations, where n_{t+1} is the dimension of the simplex (which equals the number of elements in the current basis $\mathcal{A}_{t+1}$). Since gradient projection is a descent method that maintains feasibility, it can be stopped after any number of

iterations, without prejudice to the convergence of CoGEnT .

3.1.5 Outline of the Chapter

The rest of the chapter is organized as follows. We specify CoGEnT in the next section, describing different variants of the backward step that promote parsimonious solutions (involving small numbers of atoms). In Section 3.3, we state convergence results, deferring proofs to the appendix. Section 3.4 describes the application of CoGEnT to a number of existing applications, and compares it to various other methods that have been proposed for these applications. In Section 3.5, we apply CoGEnT for a variety of *new* applications, for which current methods, if they exist at all, do not scale well to large data sets. In Section 3.6 we extend our algorithm to deal with deconvolution problems.

3.2 Algorithm

CoGEnT is specified in Algorithm 1. Its three major elements — the forward (conditional gradient) step, the backward (truncation) step, and the enhancement (reoptimization) step — have been discussed in Section 3.1. We note that these three steps are constructed so that the iterates at each step are feasible (that is, $\|\boldsymbol{x}_t\|_{\mathcal{A}} \leq \tau$). We make further notes in this section about alternative implementations of these three steps.

The forward step (Step 4) is equivalent to solving an approximation to (3.8) based on a linearization of f around the current iterate. Specifically, it is easy to show that $\tau \boldsymbol{a}_t$ solves the following problem:

$$\min_{\boldsymbol{x}} \; f(\boldsymbol{x}_t) + \langle \nabla f(\boldsymbol{x}_t), \boldsymbol{x} - \boldsymbol{x}_t \rangle \quad \text{s.t.} \quad \|\boldsymbol{x}\|_{\mathcal{A}} \leq \tau.$$

Algorithm 1 CoGEnT: Conditional Gradient with Enhancement and Truncation

1: **Input:** Characterization of $\mathcal{A}$, bound τ, acceptance threshold $\eta \in (0, 1/2]$;
2: **Initialize**, $\boldsymbol{a}_0 \in \mathcal{A}$, $t \leftarrow 0$, $\boldsymbol{A}_0 \leftarrow [\boldsymbol{a}_0]$, $c_0 \leftarrow [\tau]$, $\boldsymbol{x}_0 \leftarrow \boldsymbol{A}_0 c_0$;
3: **repeat**
4: $\quad \boldsymbol{a}_{t+1} \leftarrow \arg\min_{\boldsymbol{a} \in \mathcal{A}} \langle \nabla f(\boldsymbol{x}_t), \boldsymbol{a} \rangle$; {FORWARD}
5: $\quad \tilde{\boldsymbol{A}}_{t+1} \leftarrow [\boldsymbol{A}_t \;\; \boldsymbol{a}_{t+1}]$;
6: $\quad \gamma_{t+1} \leftarrow \arg\min_{\gamma \in [0,1]} \; f(\boldsymbol{x}_t + \gamma(\tau \boldsymbol{a}_{t+1} - \boldsymbol{x}_t))$; {LINE SEARCH}
7: $\quad \tilde{c}_{t+1} \leftarrow [(1 - \gamma_{t+1}) c_t \;\; \gamma_{t+1} \tau \boldsymbol{a}_{t+1}]$;
8: $\quad$ **Optional:** Approximately solve
$\quad \tilde{c}_{t+1} \leftarrow \arg\min_{c_{t+1}} f(\tilde{\boldsymbol{A}}_{t+1} c_{t+1})$ s.t. $\|c_{t+1}\|_1 \leq \tau, \quad c_{t+1} \geq 0$ with the output from Step 7 as a warm start; {ENHANCEMENT}
9: $\quad \tilde{\boldsymbol{x}}_{t+1} = \tilde{\boldsymbol{A}}_{t+1} \tilde{c}_{t+1}$;
10: $\quad$ Threshold $F_{t+1} := \eta f(\boldsymbol{x}_t) + (1 - \eta) f(\tilde{\boldsymbol{x}}_{t+1})$;
11: $\quad [\boldsymbol{A}_{t+1}, c_{t+1}, x_{t+1}]$
$\quad\quad = \text{TRUNCATE}(\tilde{A}_{t+1}, \tilde{c}_{t+1}, \tau, F_{t+1})$;
$\quad$ {BACKWARD}
12: $\quad t \leftarrow t + 1$;
13: **until convergence**
14: **Output:** $\boldsymbol{x}_t$

(A simple argument reveals that the minimizer of this problem is attained by $\tau \boldsymbol{a}$, where $\boldsymbol{a}$ is an atom.) For many applications of interest, this step can be performed efficiently, often more efficiently than the corresponding projection/shrinkage step in prox-linear methods.

The line search of Step 6 can be performed exactly, because of the quadratic objective in (3.8). We obtain

$$\gamma_{t+1} = \min \left\{ \frac{\langle \boldsymbol{y} - \boldsymbol{\Phi} \boldsymbol{x}_t, \boldsymbol{\Phi} \boldsymbol{v} \rangle}{\|\boldsymbol{\Phi} \boldsymbol{v}\|^2}, 1 \right\}, \; \boldsymbol{v} := \tau \boldsymbol{a}_{t+1} - \boldsymbol{x}_t.$$

We now discuss two options for performing the backward (truncation) step (Step 11), whose purpose is to compactify the representation of $\boldsymbol{x}_t$, without degrading the objective more than a specified amount. The parameter η defines a sufficient decrease criterion that the modified solution needs to satisfy. A value of η closer to its upper bound will yield more frequent

removal of atoms and hence a sparser solution, at the expense of more modest progress per iteration.

Our first implementation of the truncation step seeks to purge one or more elements from the expanded basis $\boldsymbol{A}_{t+1}$, using a quadratic prediction of the effect of removal of each atom in turn. The approach is outlined in Algorithm 2. Removal of an atom $\boldsymbol{a}$ from the current iterate $\tilde{\boldsymbol{x}}_{t+1}$ in Step 4 of Algorithm 2 would result in the following change to the objective:

$$\begin{aligned} & f(\tilde{\boldsymbol{x}}_{t+1} - c_{\boldsymbol{a}}\boldsymbol{a}) \\ & \quad = f(\tilde{\boldsymbol{x}}_{t+1}) - c_{\boldsymbol{a}}\langle \nabla f(\tilde{\boldsymbol{x}}_{t+1}), \boldsymbol{a}\rangle + \frac{1}{2}c_{\boldsymbol{a}}^2 \|\boldsymbol{\Phi}\boldsymbol{a}\|_2^2. \end{aligned} \tag{3.9}$$

(We have assumed that $c_{\boldsymbol{a}}$ is the coefficient of $\boldsymbol{a}$ in the current representation of $\tilde{\boldsymbol{x}}_{t+1}$.) The quantities $\|\boldsymbol{\Phi}\boldsymbol{a}\|_2^2$ can be computed efficiently and stored as soon as each atom $\boldsymbol{a}$ enters the current basis $\boldsymbol{A}_t$, so the main cost in evaluating this criterion is in forming the inner product $\langle \nabla f(\tilde{\boldsymbol{x}}_{t+1}), \boldsymbol{a}\rangle$. Having chosen a candidate atom that optimizes the degradation in f, we can reoptimize over the remaining elements (Step 6 in Algorithm 2), possibly using the same gradient-projection approach as in Step 8 of Algorithm 1), and test to see whether the updated value of f still falls below the threshold F_{t+1}. Note that Step 6 in Algorithm 2 is optional; we could alternately define by $\hat{c}_{t+1}$ by removing the coefficient corresponding to the discarded atom from c_{t+1}. Atom removal is repeated in Algorithm 2 as long as the successively updated objective stays below the threshold F_{t+1}.

Our second implementation of the truncation step allows for a wholesale redefinition of the current basis, seeking a new, smaller basis and a new set of coefficients such that the objecive value is not degraded too much. The approach is specified in Algorithm 3. It is motivated by the observation that atoms added at early iterates contain spurious components, which may

Algorithm 2 TRUNCATE$(\tilde{A}_{t+1}, \tilde{c}_{t+1}, \tau, F_{t+1})$

Input: Current basis $\tilde{\boldsymbol{A}}_{t+1}$, coefficient vector $\tilde{c}_{t+1}$, iterate $\tilde{\boldsymbol{x}}_{t+1} = \tilde{A}_{t+1}\tilde{c}_{t+1}$; bound τ; threshold F_{t+1};
continue $\leftarrow 1$;
while continue$= 1$ **do**
  $\hat{\boldsymbol{a}}_{t+1} \leftarrow \arg\min_{\boldsymbol{a} \in \tilde{\boldsymbol{A}}_{t+1}} f(\tilde{\boldsymbol{x}}_{t+1} - c_{\boldsymbol{a}}\boldsymbol{a})$
  $\hat{\boldsymbol{A}}_{t+1} \leftarrow \tilde{\boldsymbol{A}}_{t+1} \backslash \{\hat{\boldsymbol{a}}_{t+1}\}$;
  Find $\hat{c}_{t+1} \geq 0$ with $\|\hat{c}_{t+1}\|_1 \leq \tau$ such that $f(\hat{\boldsymbol{A}}_{t+1}\hat{c}_{t+1}) \leq f(\tilde{\boldsymbol{x}}_{t+1} - (\tilde{c}_{\tilde{\boldsymbol{a}}_{t+1}})_{t+1}\hat{\boldsymbol{a}}_{t+1})$;
  if $f(\hat{A}_{t+1}\hat{c}_{t+1}) \leq F_{t+1}$ **then**
    $\tilde{\boldsymbol{A}}_{t+1} \leftarrow \hat{\boldsymbol{A}}_{t+1}$;
    $\tilde{\boldsymbol{x}}_{t+1} \leftarrow \hat{\boldsymbol{A}}_{t+1}\hat{c}_{t+1}$;
    $\tilde{c}_{t+1} \leftarrow \hat{c}_{t+1}$;
  else
    continue $\leftarrow 0$;
  end if
end while
$\boldsymbol{A}_{t+1} \leftarrow \tilde{\boldsymbol{A}}_{t+1}$; $\boldsymbol{x}_{t+1} \leftarrow \tilde{\boldsymbol{x}}_{t+1}$; $c_{t+1} \leftarrow \tilde{c}_{t+1}$;
Output: Possibly reduced basis $\boldsymbol{A}_{t+1}$, coefficient vector $c_{t+1} \geq 0$, and iterate $\boldsymbol{x}_{t+1}$.

not be cancelled out by atoms added at later iterations. This phenomenon is apparent in matrix completion, where the number of atoms (rank-one matrices) generated by the procedure above is often considerably larger than the rank of the target matrix. For this application, we could implement Algorithm 3 by forming a singular value decomposition of the matrix represented by the latest iterate $\tilde{\boldsymbol{x}}_{t+1}$, and defining a new basis $\hat{\boldsymbol{A}}_{t+1}$ to be the rank-one matrices that correspond to the largest singular values. These singular values would then form the new coefficient vector $\hat{c}_{t+1}$, and the new iterate $\boldsymbol{x}_{t+1}$ would be defined in terms of just these singular values and singular vectors. The computational work required for such a step would be comparable with one iteration of the popular singular value thresholding (SVT) approach [11] for matrix completion, which also requires calculation of the leading singular values and singular vectors.

We conclude this section by discussing practical stopping criteria for Algorithm 1. As we

Algorithm 3 TRUNCATE$(\tilde{A}_{t+1}, \tilde{c}_{t+1}, \tau, F_{t+1})$

Input: Current basis $\tilde{\boldsymbol{A}}_{t+1}$, coefficient vector $\tilde{c}_{t+1}$, iterate $\tilde{\boldsymbol{x}}_{t+1} = \tilde{A}_{t+1}\tilde{c}_{t+1}$; bound τ; threshold F_{t+1};
Find alternative basis $\hat{\boldsymbol{A}}_{t+1}$ and coefficients $\hat{c}_{t+1} \geq 0$ such that $\#columns(\hat{\boldsymbol{A}}_{t+1}) < \#columns(\tilde{\boldsymbol{A}}_{t+1})$, $\|\hat{c}_{t+1}\|_1 \leq \tau$;
if $f(\hat{\boldsymbol{A}}_{t+1}\hat{c}_{t+1}) \leq F_{t+1}$ **then**
    $\boldsymbol{A}_{t+1} \leftarrow \hat{\boldsymbol{A}}_{t+1}$; $\boldsymbol{x}_{t+1} \leftarrow \hat{\boldsymbol{A}}_{t+1}\hat{c}_{t+1}$; $c_{t+1} \leftarrow \hat{c}_{t+1}$;
else
    $\boldsymbol{A}_{t+1} \leftarrow \tilde{\boldsymbol{A}}_{t+1}$; $\boldsymbol{x}_{t+1} \leftarrow \tilde{\boldsymbol{x}}_{t+1}$; $c_{t+1} \leftarrow \tilde{c}_{t+1}$;
end if
Output: Possibly reduced basis $\boldsymbol{A}_{t+1}$, coefficient vector $c_{t+1} \geq 0$, and iterate $\boldsymbol{x}_{t+1}$.

show in Section 3.3, CoGEnT is guaranteed to converge to an optimum, and the objective is guaranteed to decrease at each iteration. We therefore use the following termination criteria:

$$f(\boldsymbol{x}_{t+1}) \leq \text{tol}, \quad \text{or} \quad \frac{f(\boldsymbol{x}_{t-1}) - f(\boldsymbol{x}_t)}{f(\boldsymbol{x}_{t-1})} \leq \text{tol},$$

where tol is a small user-defined parameter.

3.3 Convergence Results

Convergence properties for CoGEnT are stated here, with proofs appearing in the appendix. Sublinear convergence of CoGEnT (Theorem 3.3.1) follows from a mostly familiar argument.

Theorem 3.3.1. *Consider the convex optimization problem* (3.8)*, and let* $\boldsymbol{x}^*$ *be a solution of* (3.8)*. Let* $\eta \in (0, 1/2]$*. Then the sequence of function values* $\{f(\boldsymbol{x}_t)\}$ *generated by CoGEnT converges to* $f^* = f(\boldsymbol{x}^*)$ *with*

$$f(\boldsymbol{x}_T) - f^* \leq \frac{\bar{C}}{T+1}, \quad \textit{for all } T \geq 1, \tag{3.10}$$

where

$$\bar{C}_1 := \eta D + 2(1-\eta)LR^2\tau^2,$$
$$\bar{C} := \frac{2\bar{C}_1^2}{(1-\eta)(\bar{C}_1 - LR^2\tau^2)} > 0,$$
$$L := \|\boldsymbol{\Phi}^T\boldsymbol{\Phi}\|,$$
$$D := f(\boldsymbol{x}_0) - f(\boldsymbol{x}^*),$$
$$R := \max_{\boldsymbol{a}\in\mathcal{A}} \|\boldsymbol{a}\|.$$

More interestingly, similar convergence properties hold when the atom added in the forward step of Algorithm 1 is computed *approximately*[1]. In place of the $\arg\min$ in Step 4 of Algorithm 1, we have the following requirement on $\boldsymbol{a}_{t+1} \in \mathcal{A}$:

$$\langle \nabla f(\boldsymbol{x}_t), (\tau\boldsymbol{a}_{t+1} - \boldsymbol{x}_t)\rangle \leq (1-\omega)\min_{\boldsymbol{a}\in\mathcal{A}}\langle \nabla f(\boldsymbol{x}_t), \tau\boldsymbol{a} - \boldsymbol{x}_t\rangle \tag{3.11}$$

where $\omega \in (0, 1/4)$ is a user-defined parameter. Note that (3.11) implies that $\langle \nabla f(\boldsymbol{x}_t), \tau\boldsymbol{a}_{t+1}\rangle \leq (1-\omega)\min_{\boldsymbol{a}\in\mathcal{A}}\langle \nabla f(\boldsymbol{x}_t), \tau\boldsymbol{a}\rangle + \omega\langle \nabla f(\boldsymbol{x}_t), \boldsymbol{x}_t\rangle$ so that this condition essentially requires us to find a solution of the Frank-Wolfe subproblem with relative objective accuracy ω. If a lower bound for the minimum is available from duality, this condition can be checked in practice. This criterion is similar in spirit to the inexact Newton method for nonlinear equations [62, pp. 277-279], which requires the approximate solution of the linearized model to achieve only a fraction of the decrease promised by exact solution of the model.

For the relaxed definition (3.11) of $\boldsymbol{a}_{t+1}$, we obtain the following result.

[1]Approximately solving this step can give substantial gains in practicality of the algorithm, making the method useful in a wider variety of applications, as we see in later sections

Theorem 3.3.2. *Assume that the conditions of Theorem 3.3.1 hold, but that the atom* $\boldsymbol{a}_{t+1}$ *selected in Step 4 in Algorithm 1 satisfies the condition* (3.11)*. Assume further than* $\eta \in (0, 1/3)$ *and* $\omega \in (0, 1/4)$*. Then we have*

$$f(\boldsymbol{x}_T) - f^* \leq \frac{\tilde{C}}{T+1} \quad \textit{for all } T \geq 1, \tag{3.12}$$

where

$$\tilde{C}_1 := (\eta + \omega(1-\eta))D + 2(1-\eta)LR^2\tau^2,$$
$$\tilde{C} := \frac{2\tilde{C}_1^2}{(1-\eta)[(1-\omega)\tilde{C}_1 - LR^2\tau^2]},$$

with L, R, τ, D *defined as in Theorem 3.3.1*

Finally, when the measurements obtained are noiseless, and the measurement operator $\boldsymbol{\Phi}$ has full row rank, we can prove linear convergence for CoGEnT , under Slater's condition: that is, there is a unique solution $\boldsymbol{x}^\star$ such that

$$\|\boldsymbol{x}^\star\|_{\mathcal{A}} < \tau, \quad \boldsymbol{\Phi}\boldsymbol{x}^\star = \boldsymbol{y}. \tag{3.13}$$

From [6, Proposition 3.1], we have

$$\langle \boldsymbol{r}_t, \boldsymbol{w}_t \rangle + \frac{\delta}{\sqrt{\|(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}\|}}\|\boldsymbol{r}_t\| \leq 0, \tag{3.14}$$

where

$$\delta := \text{dist}(\boldsymbol{x}^\star, \text{bdry} B_{\|\cdot\|_{\mathcal{A}}}(\tau)) = \inf_{\boldsymbol{x} \in \text{bdry} B_{\|\cdot\|_{\mathcal{A}}}(\tau)} \|\boldsymbol{x} - \boldsymbol{x}^\star\|. \tag{3.15}$$

(Note that $\delta > 0$ by assumption.) This inequality leads immediately leads to the following bounds:

$$\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2 = \|\boldsymbol{r}_t\|^2 + \|\boldsymbol{w}_t\|^2 - 2\langle \boldsymbol{r}_t, \boldsymbol{w}_t\rangle \geq \|\boldsymbol{r}_t\|^2 + \|\boldsymbol{w}_t\|^2 \geq \|\boldsymbol{w}_t\|^2, \tag{3.16a}$$

$$\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2 = \|\boldsymbol{r}_t\|^2 + \|\boldsymbol{w}_t\|^2 - 2\langle \boldsymbol{r}_t, \boldsymbol{w}_t\rangle \geq \|r_t\|^2 - \langle \boldsymbol{r}_t, \boldsymbol{w}_t\rangle = \langle \boldsymbol{r}_t, \boldsymbol{r}_t - \boldsymbol{w}_t\rangle. \tag{3.16b}$$

Recall that the closed-form expression for the optimal line-search parameter γ_t in Step 6 of Algorithm 1 is as follows:

$$\gamma_t = \min\left\{1, \frac{\langle \boldsymbol{r}_t, \boldsymbol{r}_t - \boldsymbol{w}_t\rangle}{\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2}\right\}. \tag{3.17}$$

Because of (3.16b), we have that the minimum in (3.17) is achieved at the fraction, that is,

$$\gamma_t = \frac{\langle \boldsymbol{r}_t, \boldsymbol{r}_t - \boldsymbol{w}_t\rangle}{\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2}. \tag{3.18}$$

We have the following convergence result, whose proofs tracks closely the analysis in [6].

Theorem 3.3.3. *Consider the convex optimization problem* (3.8)*, where* Φ *has full row rank. Suppose that there exists a vector* $\boldsymbol{x}^\star$ *such that* $\|\boldsymbol{x}^\star\|_{\mathcal{A}} < \tau$ *and* $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x}^\star$*. Then Co-GEnT generates a sequence of iterates* $\{\boldsymbol{x}_t\}$ *such that* $\{f(\boldsymbol{x}_t)\}$ *convergence to* $f(\boldsymbol{x}^\star) = 0$ *at a linear rate, that is,*

$$f(\boldsymbol{x}_T) \leq f(\boldsymbol{x}_0)\exp\left(-TC(1-\eta)\right), \quad \textit{where } C := \left(\frac{\delta}{\sqrt{\|(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}\|}[\|\boldsymbol{y}\| + R\tau\|\boldsymbol{\Phi}\|]}\right)^2.$$

We prove this result in Appendix A.5

3.4 Experiments: Standard Applications in Sparse Recovery

CoGEnT can be used to solve a variety of problems from signal processing and machine learning. We describe some experiences with such problems.

3.4.1 Sparse Signal Recovery

We tested our method on the following compressed sensing formulation:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}\in\mathbb{R}^p} \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}\|_2^2 \quad \text{s.t. } \|\boldsymbol{x}\|_1 \leq \tau. \tag{3.19}$$

The atoms in this case are the signed canonical basis vectors, and the atom selection (Step 4 in Algorithm 1) reduces to the following:

$$\hat{i} = \arg\max_i |[\nabla f(\boldsymbol{x}_t)]_i|,$$
$$\boldsymbol{a}_{t+1} = -\text{sign}\left([\nabla f(\boldsymbol{x}_t)]_{\hat{i}}\right) e_{\hat{i}}.$$

We consider a sparse signal $\boldsymbol{x}$ of length $p = 20000$, with 5% of coefficients randomly assigned values from $\mathcal{N}(0, 1)$. Setting $n = 5000$, we construct the $n \times p$ matrix $\boldsymbol{\Phi}$ to have i.i.d. Gaussian entries, and corrupt the measurements with Gaussian noise (AWGN) of standard deviation $\sigma = 0.01$. In the formulation (3.19), we set $\tau = \|\boldsymbol{x}^\star\|_1$, where $\boldsymbol{x}^\star$ is the chosen optimal signal.

To check the performance of CoGEnT against the conditional gradient method, we run both methods for a maximum of 5000 iterations, with a stopping tolerance of 10^{-8}. Figure 12

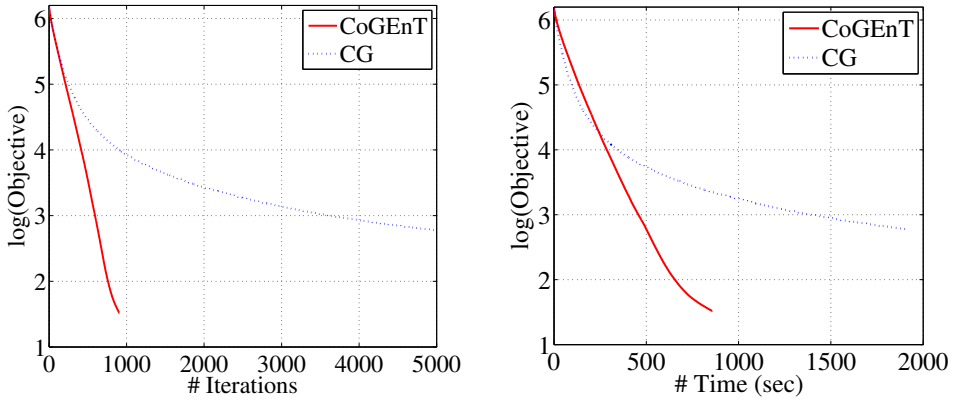


Figure 12: Comparison between CoGEnT and standard conditional gradient (CG).

shows a graph of the logarithm of the function value vs iteration count (left) and logarithm of the function value vs wall clock time (right). On a per-iteration basis, CoGEnT performs more operations than standard CG. However, the backward steps yield faster reduction in the objective function value, resulting in better convergence, even when measured in terms of run time.

Figure 13 shows a comparison of solution quality obtained by CoGEnT , CG, CoSaMP [58], and Subspace Pursuit [20]. As a performance metric, we used both the mean square error and the Hamming Distance between the true and predicted vectors. We performed 10 independent trials, setting $\mathbf{\Phi}$ in each trial to be a 1000×5000 matrix, with reference solution $\boldsymbol{x}^\star$ chosen to have $s = 200$ nonzeros. Observations $\boldsymbol{y}$ were corrupted with AWGN with standard deviation σ in the range $[0, 2]$. In CoGEnT and CG, we chose $\tau := \|\boldsymbol{x}^\star\|_1$. For CoSaMP and the Subspace Pursuit methods, we set $s = 200$, the known sparsity level of the optimal signal $\boldsymbol{x}^\star$. Figure 13 shows the results.

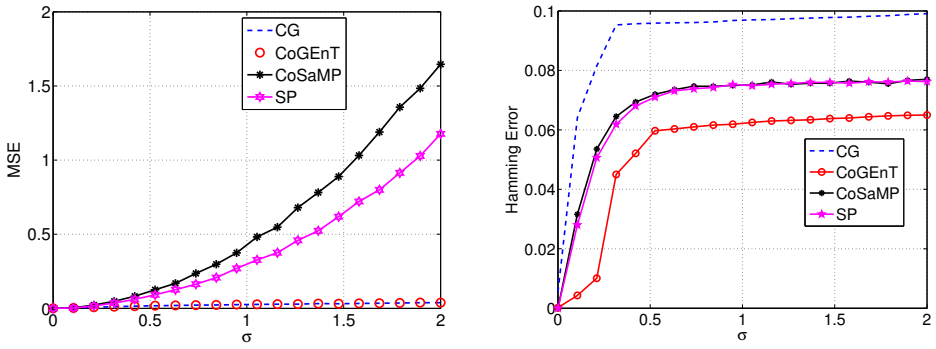


Figure 13: Comparison of solution quality obtained by different methods. Left: MSE for recovered solution as a function of observation noise parameter σ. Right: Hamming error in recovered solution as a function of σ.

3.4.2 Overlapping Group Lasso

In group-sparse variants of (3.19) we seek vectors $\boldsymbol{x}$ such that $\boldsymbol{\Phi x} \approx \boldsymbol{y}$ for given $\boldsymbol{\Phi}$ and $\boldsymbol{y}$, such that the support of $\boldsymbol{x}$ consists of a small number of predefined groups of the coefficients. We denote each group by $G \subset \{1, 2, \ldots, p\}$ and denote the full collection of groups by $\mathcal{G}$. An example of such problems is image recovery, where components of $\boldsymbol{x}$ are coefficients of discrete wavelet transforms and the groups express parent-child relationships between these coefficients. Latent group Lasso and the graph lasso [37] provide formulations and algorithms for these problems. That the penalty can be expressed as an atomic norm was shown in [68]. CG and CoGEnT approaches can be viewed as greedy analogues of the latent group lasso approach. CG and CoGEnT do not require replication of variables (as was done in [67]), and thus avoid inflating the problem dimension. The atom selection step (Step 4 in Algorithm 1) amounts to the following operation:

$$\hat{G} = \arg\max_{G \in \mathcal{G}} \| - [\nabla(f(\boldsymbol{x}_t))]_G \|,$$

$$[\boldsymbol{a}_{t+1}]_{\hat{G}} = -[\nabla f(\boldsymbol{x}_t))]_{\hat{G}} / \| [\nabla f(\boldsymbol{x}_t))]_{\hat{G}} \|,$$

$$[\boldsymbol{a}_{t+1}]_i = 0 \ \text{ for } i \notin \hat{G}.$$

We test the CoGEnT and CG approaches on some standard one-dimensional signals from [17], aiming to recover DWT coefficients arranged into parent-child groups. We take $\boldsymbol{\Phi}$ to be an $n \times p$ Gaussian matrix, with $p = 1024$ and $n = 300$. The entries of the measurement vector $\boldsymbol{y}$ are corrupted with AWGN of standard deviation $\sigma = 0.01$. Each signal was scaled to lie between 0 and 1, and we restricted ourselves to 200 iterations of each algorithm. Table 3 shows final MSE values for the methods. Note that in all cases, the CG method selects 200 atoms (equal to the number of iterations), while CoGEnT selects far fewer atoms (see final column) while producing somewhat closer MSE fits to the ground truth.

Signal	**MSE CoGEnT**	**MSE CG**	**#Atoms Selected**
Piece Polynomial	1.262×10^{-4}	2.267×10^{-4}	44
Blocks	5.933×10^{-5}	1.129×10^{-4}	52
HeaviSine	5.164×10^{-4}	7.118×10^{-4}	64
Piecewise Regular	0.0017	0.0083	62

Table 3: Recovery of some 1d test signals in the presence of AWGN ($\sigma = 0.01$). After 200 iterations, ECG recovers more accurate and sparser solutions.

We conpare next the performance of CoGEnT in comparision with the latest group Lasso (LGL) approach, the latter using replication of variables that appear in multiple groups. We considered M group sparse signals with $\lfloor M/10 \rfloor$ groups chosen to be active in the reference solution, where each group has size 50. The groups are ordered in linear fashion with the

last 30 indices of each group overlapping with the first 30 of the next group. We then took $n = \lceil p/2 \rceil$ measurements with a Gaussian sensing matrix $\mathbf{\Phi}$, with AWGN of standard deviation $\sigma = 0.1$ added to the observations. Table 4 shows runtimes for CoGEnT and LGL, the latter implemented using SpaRSA [90] on the formulation with replicated variables. (It is possible to implement LGL without explicitly replicating columns of $\mathbf{\Phi}$, but we found for these experiments that there was little computational advantage in doing so.)

M	True Dimension	Replicated Dimension	time CoGEnT	time LGL
100	2030	5000	15.	22.
1000	20030	50000	211.	462.
1200	24030	60000	359.	778.
1500	30030	75000	575.	1377.
2000	40030	100000	852.	2977.

Table 4: Recovery times (in seconds) for CoGEnT and latent group Lasso (LGL) applied to a synthetic group-sparse problem.

3.4.3 Group ℓ_1-ℓ_∞ Regularization for Multitask Learning

In multitask learning applications, we desire not only for the covariates to be shared across many tasks, but also that they share the same magnitude of activation. Such problems, and their use in other applications, have been considered in [50, 60] and elsewhere. In such cases, a group ℓ_1-ℓ_∞ regularizer is used to encourage the desired sparsity pattern. Such norms can be easily cast into the atomic norm framework by defining the atoms be the unit ℓ_∞ ball restricted to a given group. Step 4 in Algorithm 1 is then identical to the step for the latent group lasso, with the ℓ_2 norm replaced by the ℓ_1 norm (the dual of the ℓ_∞ norm) to select $\hat{G}$.

We define a synthetic problem with 1000 features and 5 tasks. We consider a Gaussian sensing matrix with $n = 350$ rows, common to all tasks. (The observations then form a 350×5 matrix.) We generate a feature matrix $\boldsymbol{X}^\star \in \{\pm 1, 0\}^{1000 \times 5}$ by choosing 20 rows at random,

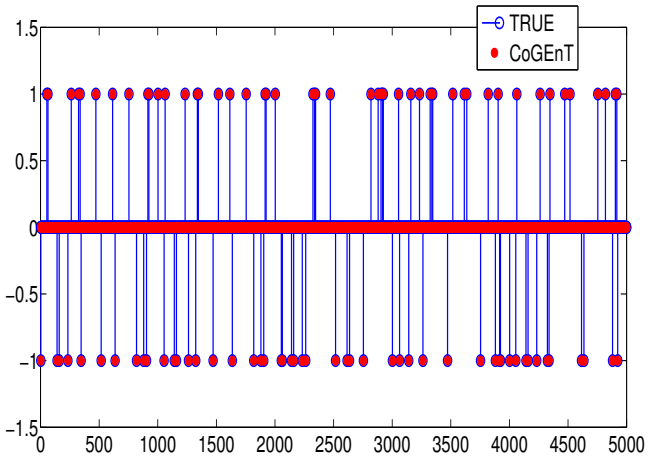


Figure 14: CoGEnT for multitask learning with the ℓ_1-ℓ_∞ norm regularizer. (The figure shows a vectorized 1000×5 matrix.) Final MSE is 0.000009 and we obtain perfect signed support recovery.

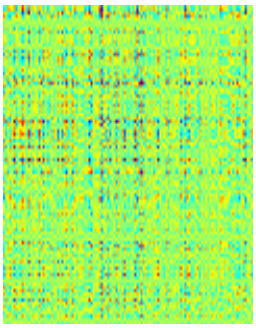
and populating each such row with Rademacher(0.5) random variables. The measurements are corrupted by AWGN with standard deviation $\sigma = 0.02$.

Figure 14 shows the result obtain by CoGEnT , with matrices vectorized for the sake of display. Note that the recovery is essentially perfect.

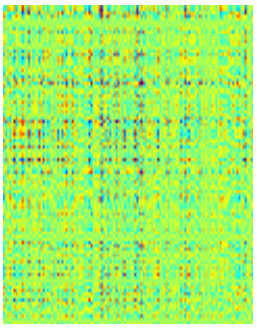
3.4.4 Matrix Completion

In low-rank matrix completion, the atoms are rank-one matrices and the observations are individual elements of the matrix. We generated a synthetic 100×120 random matrix with rank $r = 3$, and observed only 30% of its entries in randomly chosen locations. We set the parameter τ to its empirically optimal value - the one that gave best results for recovery. We follow up the CoGEnT and CG algorithms with a debiasing step, in which the bound involving τ is discarded and we solve a least-squares problem over the final basis of rank-one matrices to identify a set of nonnegative coefficients that fits the observations best. For the backward step,

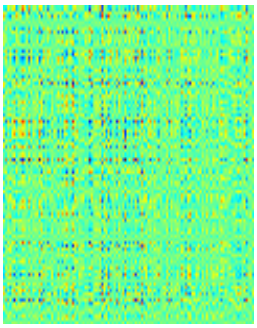
we use Algorithm 3. We see in Figure 15 that CoGEnT recovers the original matrix well; the three singular values are recovered almost exactly. By contrast, CG gives a solution with five nonzero singular values.



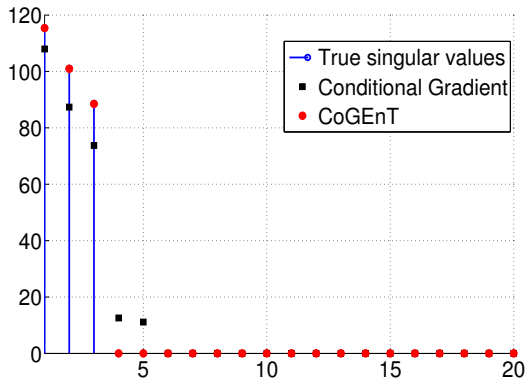
(a) Original Matrix of rank 3



(b) Matrix recovered from CoGEnT



(c) Matrix recovered from conditional gradient



(d) Singular values of recovered matrices and ground truth

Figure 15: Matrix completion using CoGEnT and CG. Note that CoGEnT recovers the true matrix almost exactly and identifies the rank correctly.

We compare CoGEnT to OptSpace [45] and SET [21] on larger problems. We vary the number of rows m of the target matrix, setting the number of columns $n = \lceil \frac{4}{3}m \rceil$. We generate a matrix of size $m \times n$, having rank $r = \max\left\{2, \lceil \frac{m}{100} \rceil\right\}$. We observe 25% of the entries at random, and corrupt the measurements with Additive White Gaussian Noise of standard

deviation 0.02. We then aim to recover the matrix using OptSpace and CoGEnT . We set the maximum number of iterations to be 100, and the tolerance to be 10^{-6}. Figure 16 plots the time taken to run each method, as a function of the number of rows of the matrix m. Each point on the curves is a result of averaging over 10 independent runs.

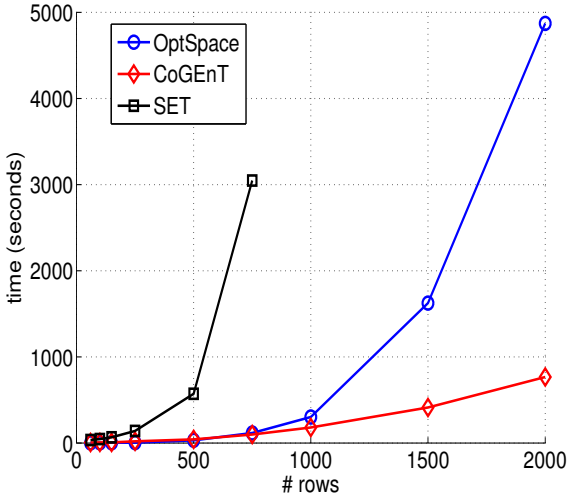


Figure 16: Time taken to run CoGEnT , OptSpace and SET. As the size of the matrix increases, CoGEnT takes far less time than OptSpace. SET does not scale too well as the matrix size increases, and we did not run it beyond matrices with 750 rows

3.5 Experiments: Novel Applications

We now report on the application of CoGEnT to recovery problems in several novel areas of application. In some cases, CoGEnT and CG are the only practical approaches for solving these problems.

3.5.1 Tensor Completion

Recovery of low-rank tensor approximations arises in applications ranging from multidimensional signal processing to latent-factor models in machine learning [1]. We consider the recovery of symmetric orthogonal tensors from incomplete measurements using CoGEnT . We seek a tensor T of the form $T = \sum_{i=1}^{r} c_i[\otimes \boldsymbol{u}_i]$, where $\otimes \boldsymbol{u}$ indicates an t-fold tensor product of a vector $\boldsymbol{u} \in \mathbb{R}^p$. We obtain partial measurements of this tensor of the form $y = \mathcal{M}(T)$, where $\mathcal{M}(\cdot)$ is a *masking operator* that reveals a certain subset of the entries of the tensor. We formulate this problem in an atomic norm setup, wherein the objective function that captures the data fidelity term is $f(T) := \frac{1}{2}\|y - \mathcal{M}(T)\|^2$. The atomic set has the form

$$\mathcal{A} = \{\otimes \boldsymbol{u} : \boldsymbol{u} \in \mathbb{R}^p,\ \|\boldsymbol{u}\|_2 = 1\}.$$

In applying CoGEnT to this problem, the greedy step requires calculation of the symmetric rank-one tensor that best approximates the gradient of the loss function. This calculation can be performed efficiently using power iterations [1]. We implement a backward step based on basis reoptimization and thresholding (Algorithm 3), where the new basis is obtained from a tensor decomposition, computed via power iterations.

We look to recover toy $10 \times 10 \times 10$ tensors, with 50% of the entries observed using CoGEnT (without noise). Figure 5 shows accuracy of recovery for tensors of various ranks. While the recovered tensor does not always match the rank of the original tensor, it does indeed have low rank and small component-wise error. We declare that recovery is “exact" if each entry of the recovered tensor is within 10^{-3} relative error w.r.t. the original tensor. We used a (relative) stopping tolerance of 10^{-6}, running the method for a maximum of 100 iterations.

Fig. 17 shows the probability of successfully performing tensor recovery for random $20 \times$

Rank	MSE
2	5.406×10^{-5}
3	3.4789×10^{-4}
4	4.999×10^{-5}
5	5.4929×10^{-4}

Table 5: Accuracy of tensors recovered, from 50% of exact observations.

20×20 tensors, using different fractions of sampled entries. Compared to the matrix unfolding method, we see that CoGEnT requires far fewer entries to perform accurate recovery. The results were averaged over 10 independent trials.

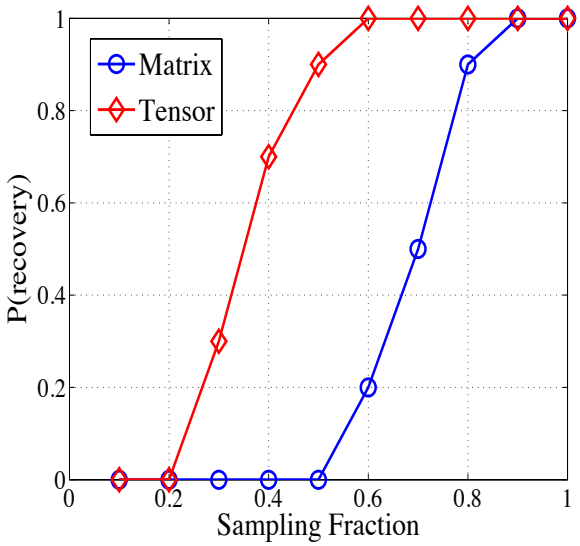


Figure 17: Recovery vs fraction of observations. Notice that the matrix unfolding method requires far more observations.

3.5.2 Moment Problems in Signal Processing

Consider a continuous time signal

$$\phi(t) = \sum_{j=1}^{k} c_j \exp(i2\pi f_j t),$$

for frequencies $f_j \in [0,1]$, $j = 1, 2, \ldots, k$ and coefficients $c_j > 0$, $j = 1, 2, \ldots, k$. In many applications of interest, $\phi(t)$ is sampled at times $S := \{t_i\}_{i=1}^n$ giving an observation vector $\boldsymbol{x} := [\phi(t_1), \phi(t_2), \ldots, \phi(t_n)] \in \mathbb{C}^n$. The observed information is therefore

$$\boldsymbol{x} = \sum_{j=1}^{k} c_j a(f_j),$$

where

$$a(f_j) = \left[e^{i2\pi f_j t_1}, e^{i2\pi f_j t_2}, \ \ldots, e^{i2\pi f_j t_n}\right]^T.$$

Finding the unknown coefficients c_j and frequencies f_j from $\boldsymbol{x}$ is a challenging problem in general. A natural convex relaxation, analyzed in [81], is obtained by setting $\boldsymbol{\Phi} = I$ in (3.8) and defining the atoms to be $a(f)$ for $f \in [0,1]$, a set of infinite cardinality.

The main technical issue in applying CoGEnT to this problem is the greedy atom selecion step (Step 4 of Algorithm 1), which requires us to find the maximum modulus of a trigonometric polynomial on the unit circle. This operation can be formulated as a semidefinite program [28], but since SDPs do not scale well to high dimensions [81], this approach has limited appeal. In our implementation of CoGEnT , we form a discrete grid of frequency values. We start with an initial grid of equally spaced frequencies, then refine it between iterations by adding new frequencies midway between each pair of selected frequencies.

By controlling the discretization in this way, we are essentially controlling the inexactness of the forward step. Indeed, the accuracy requires in (3.11) can provide guidance for the adaptive discretization process. Step 4 simply selects an atom $a(f)$ corresponding to the frequency f in the current grid that forms the most negative inner product with the gradient of the loss function.

Our implementation of the backward step for this problem has two parts. Besides performing Algorithm 2 to remove multiple uninteresting frequencies, we include a heuristic for merging nearby frequencies, replacing multiple adjacent spikes by a single spike, when it does not degrade the fit to observations too much to do so.

Figure 18 compares the performance of CoGEnT with that of standard CG on a signal with ten randomly chosen frequencies in $[0, 1]$. We take samples at 300 timepoints of a signal of length 1000, corrupted with AWGN with standard deviation $.01$. The left figure in Figure 18 shows the signal recovered by CoGEnT , indicating that all but the smallest of the ten spikes were recovered accurately. The critical role played by the backward step can be seen by contrasting these results with those reported for CG in the right figure of Fig. 18, where many spurious frequencies appear.

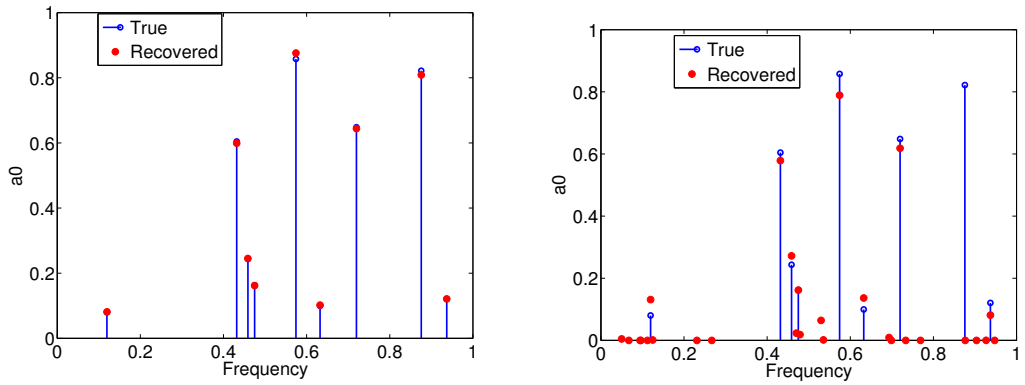


Figure 18: CoGEnT and CG for off-grid compressed sensing. Blue spikes and circles represent the reference solution, and red circles are those estimated by the algorithms.

We compared CoGEnT to the SDP formulation as explained in [81]. Although the SDP solves the problem exactly, it does not scale well to large dimensions, as we show in the timing comparisons of Figure 19.

The formulation above can be generalized to include signals that are a conic combination

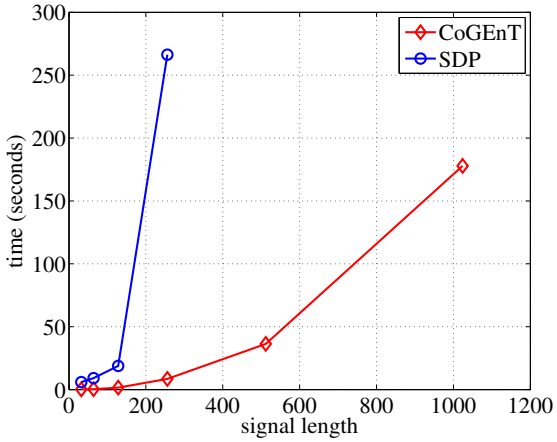


Figure 19: Speed comparison with SDP. The SDP formulation does not scale well, and we could not run it for signal sizes beyond 256.

of a few arbitrary functions of the form $\phi(t, \alpha_i)$.

- Bessel and Airy functions form natural signal ensembles that arise as solutions to differential equations in physics. As an example, letting $J_r(t)$ denoting Bessel functions of the first kind, we have

$$\phi(t; \alpha_1, \alpha_2, \alpha_3) = J_{\alpha_1}\left(\frac{t}{\alpha_2} - \alpha_3\right),$$

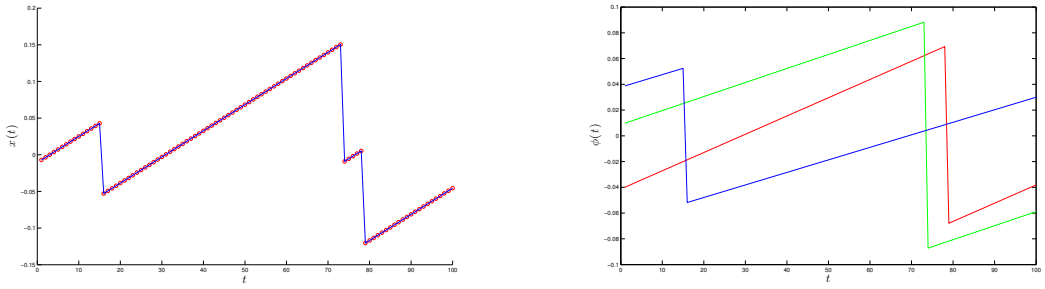
  where $\alpha_1, \alpha_2, \alpha_3 \in \mathbb{R}_+$. Here, each atom is defined by a specific choice of the triple $(\alpha_1, \alpha_2, \alpha_3)$. (Again, the atomic set $\mathcal{A}$ has infinite cardinality.)

- Triangle and sawtooth waves. Consider for instance the sawtooth functions:

$$\phi(t; \alpha_1, \alpha_2) = \frac{t}{\alpha_1} - \left\lfloor \frac{t}{\alpha_1} \right\rfloor - \alpha_2,$$

  where $\alpha_1, \alpha_2 \in \mathbb{R}_+$. Each atom is defined by a specific choice of (α_1, α_2). Figure 20

shows successful recovery of a superposition of sawtooth functions from a limited number of samples.



(a) The true signal (blue) is a superposition of sawtooth functions. Red dots show samples acquired.

(b) Sawtooth components recovered by CoGEnT .

Figure 20: Recovering sawtooth components by sampling. (Best seen in color)

- Ricker wavelets arise in seismology applications, with the atoms characterized by $\sigma > 0$:

$$\phi(t;\sigma) = \frac{2}{\sqrt{3\sigma}\pi^{\frac{1}{4}}}\left(1-\frac{t^2}{\sigma^2}\right)\exp\left(-\frac{t^2}{2\sigma^2}\right).$$

- Gaussians, characterized by parameters μ and σ:

$$\phi(t;\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(t-\mu)^2}{2\sigma^2}\right).$$

  Estimating Gaussian mixtures from sampled data is a much-studied problem in machine learning.

The key ingredient in solving these problems within the atomic norm framework is efficient (approximate) solution of the atom selection step. In some cases, this can be done in closed form, whereas for all the signals mentioned above, approximate solutions can be obtained via adaptive discretization.

3.5.3 Group Testing on Graphs

We apply CoGEnT for group testing on graphs, where the atoms can either correspond to nodes in a graph, edges, or cliques. Assuming the nodes in a graph are variables, we say that a node (or a set of nodes) in a graph is active if it represents a variable that is relevant to the task at hand. This could correspond to identifying faulty sensors in a network, or identifying a clique in a social network. For identifying cliques, we merely need to solve a group-lasso version of the graph testing methods considered in [18], as explained below:

We model the problem as an atomic norm based recovery problem. Let $\mathcal{N} = \{1, 2, \ldots, N\}$ be the set of nodes and $\mathcal{E} \subset \mathcal{N} \times \mathcal{N}$ be the set of (undirected) edges. The activation pattern that we seek to recover is a vector $\boldsymbol{x}^\star$, whose components represent the activation values associated with each node. For each $n \in \mathcal{N}$, we define the group g_n of neighbors of node n, that is,

$$g_n := \{x_i : (i, n) \in \mathcal{E}\}, \quad i = 1, 2, \ldots, N.$$

We then form a set of groups as

$$\mathcal{G} = \{g_n \, : \, n = 1, 2, \ldots, N\}.$$

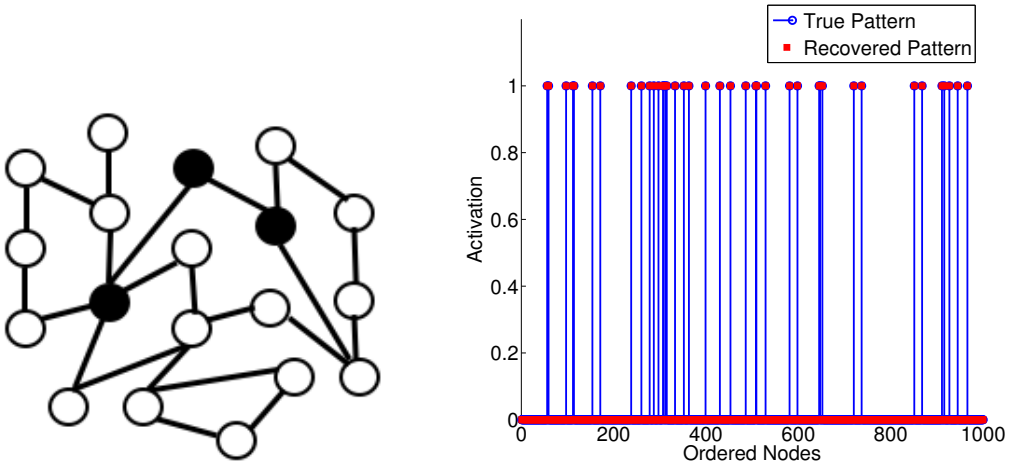
Consider the following atomic set:

$$\mathcal{A} := \bigcup_{n=1,2,\ldots,N} \{\boldsymbol{a} \in \mathbb{R}^N \, : \, \boldsymbol{a}_i \in \{-1, +1\} \text{ if } i \in g_n\}.$$

In the sets in the union, $\boldsymbol{a}_i = 0$ if $i \notin g_n$. The atomic norm induced by this set admits a solution that can be decomposed into active groups, which in this case will be neighborhoods

of nodes. The atomic norm will also force the components of $\boldsymbol{x}$ that correspond to active nodes to have the same magnitude. The atom selection step in this case then reduces to that for the $\ell_1 - \ell_\infty$ multitask learning considered in the previous section.

To test the performance of CoGEnT on these problems, we constructed a synthetic graph of $N = 1000$ nodes. The edge set was defined randomly to have a density of about 10%. To form the reference solution $\boldsymbol{x}^\star$, we chose a node $n \in \mathcal{N}$ arbitrarily at random to have the value 1, and set its neighbors to have value 1 also, while the components of $\boldsymbol{x}$ corresponding to all other nodes were set to 0. (Note that this solution is covered by the single group g_n.) We obtained 300 measurements from this graph using a random bernoulli (0.5) sensing matrix. Figure 21b confirms that CoGEnT performs perfect recovery of $\boldsymbol{x}^\star$.



(a) The reference solution is defined to be a single node and its neighbors (shaded).

(b) Solution recovered by CoGEnT on a random graph with 1000 nodes and density 10%.

Figure 21: Problem and recovery results for CoGEnT applied to recovery of graph activation patterns

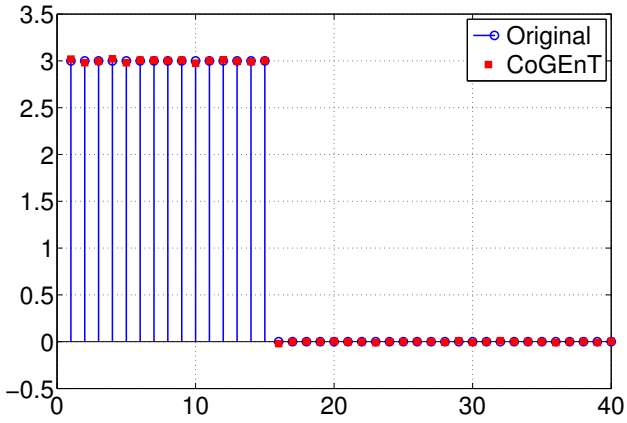


Figure 22: Recovery of a vector with correlated variables obtained by applying CoGEnT to OSCAR

3.5.4 OSCAR

The regularizer for the Octagonal Shrinkage and Clustering Algorithm for Regression (OSCAR) method is defined for $\boldsymbol{x} \in \mathbb{R}^p$ as follows:

$$\|\boldsymbol{x}\|_1 + c \sum_{j=1}^{p} \sum_{k=1}^{j} \max\left\{|\boldsymbol{x}_j|, |\boldsymbol{x}_k|\right\}$$

The atomic-norm formulation is obtained by defining the atoms to be the vectors with at most two non zero entries, resulting in a scaled ℓ_∞ norm ball in 2D and the ℓ_1 norm ball in the ambient space. We considered the example of [9, Section 4, Example 5], corrupting the measurements with AWGN of standard deviation 0.05. CoGEnT was used to recover the reference vector β, varying the bound τ and choosing the value that performed best. Figure 22 shows that CoGEnT succeeds in recovering the solution.

3.5.5 Successive Projections for Tree-Structured Norms

Hierarchical structured sparsity penalties find application in a variety of signal processing and machine learning applications [43]. Problems of this form can be written as a tree-structured group lasso. This framework ca be extended to handle overlapping groups in a level as well [66]. We show that CoGEnT lends itself well to solve problems of this form in an efficient manner.

To keep the exposition simple, we consider two levels in the hierarchy: a set of groups and singletons, corresponding to the sparse group lasso formulation. We define two sets of atoms, one for each level of the hierarchy. For the upper (group) level, the atoms correspond to vectors with support restricted to a group $G \in \mathcal{G}$. We first pick an atom $\boldsymbol{a}_{upper}$ from this set of atoms, by performing the atom selection step as done in the latent group lasso formulation.

Once we select an atom corresponding to the upper level of the hierarchy, we then define another set of atoms: canonical basis vectors $\boldsymbol{e}_i$ but only for $i \in supp(\boldsymbol{a}_{upper})$, the support of the atom selected in the upper level. This results in a greedy step of the following form:

$$\mathcal{A}_{upper} = \bigcup_{G \in \mathcal{G}} \{\boldsymbol{a} : \boldsymbol{a}_{\setminus G} = 0, \|\boldsymbol{a}\|_2 = 1\}$$

$$\boldsymbol{a}_{upper} = \arg \max_{\boldsymbol{a} \in \mathcal{A}_{upper}} \langle \boldsymbol{a}, -\nabla f_t \rangle$$

$$\mathcal{A}_{lower} = \{\boldsymbol{e}_i : i \in \text{supp}(\boldsymbol{a}_{upper})\}$$

$$\boldsymbol{a} = \arg \max_{\boldsymbol{a} \in \mathcal{A}_{lower}} \langle \boldsymbol{a}, -\nabla f_t \rangle$$

An important thing to note here is that the efficiency of the method is not affected by overlapping hierarchical groups, as considered in the SOS lasso framework.

To test this approach, we considered 100 groups of size 20, resulting in a signal of length

2000. We selected 10 groups at random, and set them to be non-zero. Among the active groups, only 15% of the coefficeints were active. We obtained 667 linear measurements. Figure 23 shows the results.

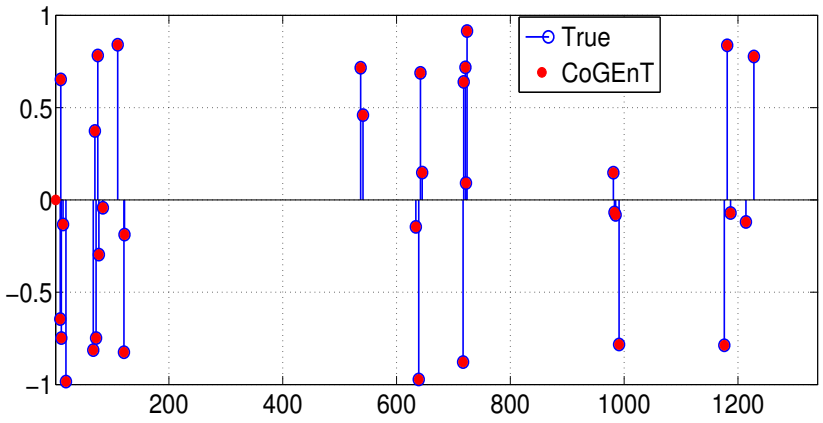


Figure 23: Recovery of a Tree group structured signal.

Note that in this case, the atom selection step is approximately solved. The solution of CoGEnT converges to the true solution only if the approximation is “good enough". Although experimentally we see that we do obtain accurate solutions, an interesting theoretical endeavor would be to characterize the goodness of such approximations.

3.6 Reconstruction and Deconvolution

The deconvolution problem involves recovering a signal of the form $\boldsymbol{x} = \boldsymbol{x}^1 + \boldsymbol{x}^2$ from observations $\boldsymbol{y}$ via a sensing matrix $\boldsymbol{\Phi}$, where $\boldsymbol{x}^1$ and $\boldsymbol{x}^2$ can be expressed compactly with respect to different atomic sets $\mathcal{A}_1$ and $\mathcal{A}_2$.

We have

$$\boldsymbol{x}^1 = \sum_{\boldsymbol{a} \in \mathcal{A}_1} c_{\boldsymbol{a}}^1 \boldsymbol{a}, \qquad \boldsymbol{x}^2 = \sum_{\boldsymbol{a} \in \mathcal{A}_2} c_{\boldsymbol{a}}^2 \boldsymbol{a},$$

where only a small number of atoms are present in each expansion.

We mentioned several instances of such problems in Section 3.1. Adopting the optimization-driven approach outlined in Section 3.1, we arrive at the following convex optimization formulation:

$$\begin{aligned} \underset{\boldsymbol{x}^1, \boldsymbol{x}^2}{\text{minimize}} \quad & \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{\Phi}(\boldsymbol{x}^1 + \boldsymbol{x}^2)\|^2 \\ \text{subject to} \quad & \|\boldsymbol{x}^1\|_{\mathcal{A}_1} \leq \tau_1 \text{ and } \|\boldsymbol{x}^2\|_{\mathcal{A}_2} \leq \tau_2. \end{aligned}$$

Algorithm 1 can be extended to this situation, as we describe informally now. Each iteration starts by choosing an atom from $\mathcal{A}_1$ that nearly minimizes its inner product with the gradient of the objective function with respect to $\boldsymbol{x}_1$; this is the forward step with respect to $\mathcal{A}_1$. One then performs a backward step for $\mathcal{A}_1$. Next follows a similar forward step with respect to $\mathcal{A}_2$, followed by a backward step for $\mathcal{A}_2$. We then proceed to the next iteration, unless convergence is flagged. Note that the backward steps are taken only if they do not deteriorate the objective function beyond a specified threshold. The entire procedure is repeated until a termination condition is satisfied.

In our first example, we consider the standard recovery of sparse + low rank matrices. We consider a matrix of size 50×50, which is a sum of a random rank 4 matrix and a sparse matrix with 100 entries. The sets $\mathcal{A}_1$ and $\mathcal{A}_2$ are defined in the usual way for these types of matrices. Figure 24 shows the true components, and Figure 25 shows that CoGEnT recovers

the components accurately

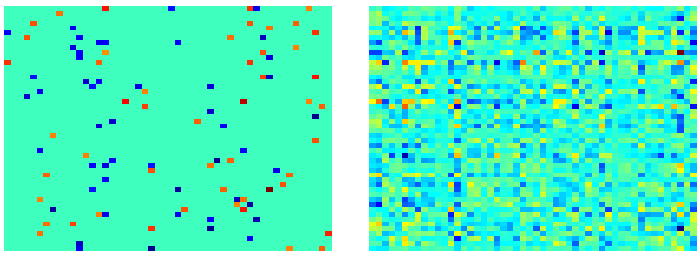


Figure 24: True sparse and low rank components

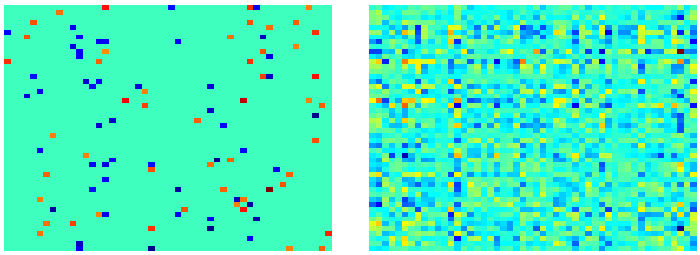


Figure 25: Recovered sparse and low rank components when the true ones are those shown in Figure 24, Error in each recovered component is at most 10^{-7}.

We also consider recovery of a mixture of signals that are sparse in the canonical and DCT bases. We generated random signals with sparsity level 10 in each of the bases, and applied our method to perform recovery. Figure 26 shows that our method indeed recovers the components accurately

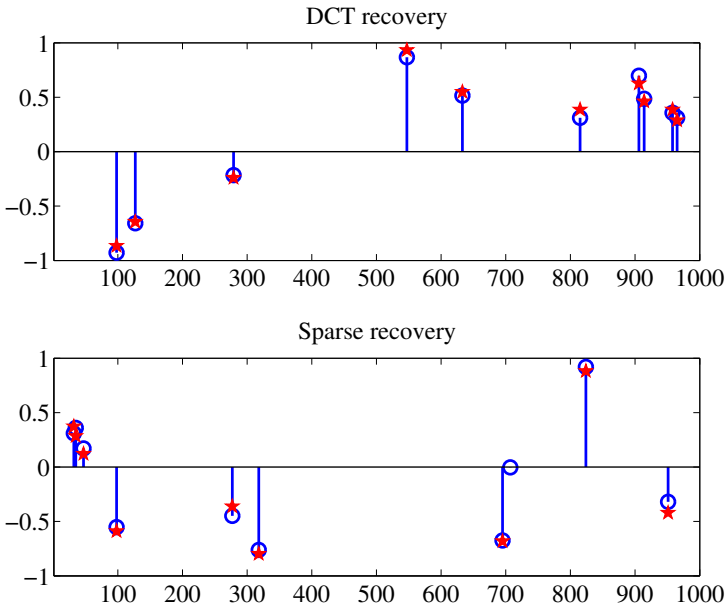


Figure 26: Recovery of a signal that is sparse in the DCT and canonical basis. The MSE for the top figure is 2.3×10^{-5}, and that for the lower figure is 3.3×10^{-5}. The blue bars represent the true components and the red stars represent the recovered coefficients

3.7 Conclusions

In this chapter, we introduced CoGEnT , a greedy scheme for recovering signals that are representable as a linear combination of a few elements from some basis. We showed that our method is efficient, and helps in obtaining solutions that are sparse in the bases of interest. CoGEnT enjoys the same theoretical convergence properties as conditional gradient and is applicable in a variety of interesting problems, including compressed sensing, matrix completion, and moment problems. We also extended the method to problems in signal demixing.

Chapter 4

Applications in Structured Sparse Signal Recovery

In Chapter 2, we focused on theoretical characterization of the SOSlasso and the group lasso. In this chapter, we exclusively focus on applications involving the SOSlasso and the group lasso. We start with an application in multi-subject fMRI that motivated work on the SOSlasso, in a multitask learning framework. We then move on to an application in computational biology, where the groups are predefined, and the goal is to identify the relevant genes for metastasis of breast cancer tumors.

We then turn our attention to a novel method to model wavelet sparsity coefficients in inverse problems in image processing. Our method allows us to recover wavelet sparsity patterns using convex optimization framework, while at the same time taking advantage of the structure inherent among the coefficients. We also discuss a method to design measurement matrices for compressed sensing applications, that takes advantage of this structure. We show that by modifying the measurements to take advantage of the structure, we can achieve further gains over the standard lasso and group lasso based methods for compressive imaging

4.1 The Sparse Overlapping Sets lasso for Multitask Learning in fMRI Applications

The SOS lasso introduced in Chapter 2 is motivated in part by multitask learning applications. The group lasso is a commonly used tool in multitask learning, and it encourages the same set of features to be selected across all tasks. As mentioned before, we wish to focus on a less restrictive version of multitask learning, where the main idea is to encourage sparsity patterns that are similar, but not identical, across tasks. Such a restriction corresponds to a scenario where the different tasks are related to each other, in that they use similar features, but are not exactly identical. This is accomplished by defining subsets of similar features and searching for solutions that select only a few subsets (common across tasks) and a sparse number of features within each subset (possibly different across tasks). Figure 27 shows an example of the patterns that typically arise in sparse multitask learning applications, along with the one we are interested in. We see that the SOSlasso, with it's ability to select a few groups and only a few non zero coefficients within those groups lends itself well to the scenario we are interested in.

A major application that we are motivated by is the analysis of multi-subject fMRI data, where the goal is to predict a cognitive state from measured neural activity using voxels as features. Because brains vary in size and shape, neural structures can be aligned only crudely. Moreover, neural codes can vary somewhat across individuals [30]. Thus, neuroanatomy provides only an approximate guide as to where relevant information is located across individuals: a voxel useful for prediction in one participant suggests the general anatomical neighborhood where useful voxels may be found, but not the precise voxel. Past work in inferring sparsity

(a) Sparse (b) Group sparse (c) Group sparse *plus* sparse (d) Group sparse *and* sparse

Figure 27: A comparison of different sparsity patterns in the multitask learning setting. Figure (a) shows a standard sparsity pattern. An example of group sparse patterns promoted by Glasso [91] is shown in Figure (b). In Figure (c), we show the patterns considered in [40]. Finally, in Figure (d), we show the patterns we are interested in this chapter. The groups are sets of rows of the matrix, and can overlap with each other

patterns across subjects has involved the use of groupwise regularization [87], using the logistic lasso to infer sparsity patterns without taking into account the relationships across different subjects [74], or using the elastic net penalty to account for groupings among coefficients [71]. These methods do not exclusively take into account both the common macrostructure and the differences in microstructure across brains, and the SOSlasso allows one to model both the commonalities and the differences across brains. Figure 28 sheds light on the motivation, and the grouping of voxels across brains into overlapping sets

In the multitask learning setting, suppose the features are give by $\mathbf{\Phi}_t$, for tasks $t = \{1, 2, \ldots, \mathcal{T}\}$, and corresponding sparse vectors $\boldsymbol{x}_t^\star \in \mathbb{R}^p$. These vectors can be arranged as columns of a matrix $\boldsymbol{X}^\star$. Suppose we are now given M groups $\tilde{\mathcal{G}} = \{\tilde{G}_1, \tilde{G}_2, \ldots\}$ with maximum size $\tilde{B}$. Note that the groups will now correspond to sets of rows of $\boldsymbol{X}^\star$.

Let $\boldsymbol{x}^\star = [\boldsymbol{x}_1^{\star T} \quad \boldsymbol{x}_2^{\star T} \quad \ldots \boldsymbol{x}_{\mathcal{T}}^{\star T}]^T \in \mathbb{R}^{\mathcal{T}p}$, and $\boldsymbol{y} = [\boldsymbol{y}_1^T \quad \boldsymbol{y}_2^T \quad \ldots \boldsymbol{y}_{\mathcal{T}}^T]^T \in \mathbb{R}^{\mathcal{T}n}$. We also define $\mathcal{G} = \{G_1, G_2, \ldots, G_M\}$ to be the set of groups defined on $\mathbb{R}^{\mathcal{T}p}$ formed by aggregating the rows of $\boldsymbol{X}$ that were originally in $\tilde{\mathcal{G}}$, so that $\boldsymbol{x}$ is composed of groups $G \in \mathcal{G}$, and let

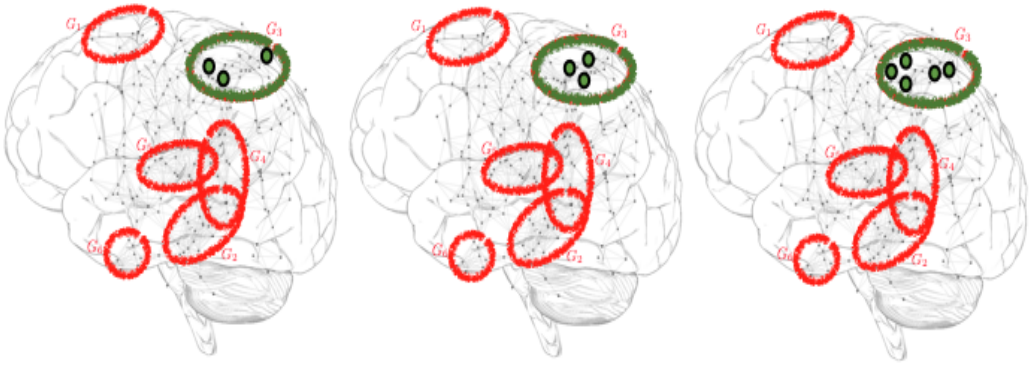


Figure 28: SOSlasso for fMRI inference. The figure shows three brains, and voxels in a particular anatomical region are grouped together, across all individuals (red and green ellipses). For example, the green ellipse in the brains represents a single group. The groups denote anatomically similar regions in the brain that may be co-activated. However, within activated regions, the exact location and number of voxels may differ, as seen from the green spots.

the corresponding maximum group size be $B = \mathcal{T}\tilde{B}$. By organizing the coefficients in this fashion, we can reduce the multitask learning problem into the standard form as considered in Chapter 2.

Results on fMRI dataset

In this experiment, we compared SOSlasso, lasso, standard multitask group lasso (with each feature grouped across tasks), the overlapping group lasso [37] (with the same groups as in SOSlasso) and the Elastic Net [95] in analysis of the star-plus dataset [88]. 6 subjects made judgements that involved processing 40 sentences and 40 pictures while their brains were scanned in half second intervals using fMRI[1]. We retained the 16 time points following each stimulus, yielding 1280 measurements at each voxel. The task is to distinguish, at each point

[1]Data and documentation available at http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-81/www/

in time, which kind of stimulus a subject was processing. [88] showed that there exists cross-subject consistency in the cortical regions useful for prediction in this task. Specifically, experts partitioned each dataset into 24 non overlapping regions of interest (ROIs), then reduced the data by discarding all but 7 ROIs and, for each subject, averaging the BOLD response across voxels within each ROI. With the resulting data, the authors showed that a classifier trained on data from 5 participants generalized above chance when applied to data from a 6th–thus proving some degree of consistency across subjects in how the different kinds of information were encoded.

We assessed whether SOSlasso could leverage this cross-individual consistency to aid in the discovery of predictive voxels without requiring expert pre-selection of ROIs, or data reduction, or any alignment of voxels beyond that existing in the raw data. Note that, unlike [88], we do not aim to learn a solution that generalizes to a withheld subject. Rather, we aim to discover a group sparsity pattern that suggests a similar set of voxels in all subjects, before optimizing a separate solution for each individual. If SOSlasso can exploit cross-individual anatomical similarity from this raw, coarsely-aligned data, it should show reduced cross-validation error relative to the lasso applied separately to each individual. If the solution is sparse within groups and highly variable across individuals, SOSlasso should show reduced cross-validation error relative to Glasso. Finally, if SOSlasso is finding useful cross-individual structure, the features it selects should align at least somewhat with the expert-identified ROIs shown by [88] to carry consistent information.

We trained the 5 classifiers using 4-fold cross validation to select the regularization parameters, considering all available voxels without preselection. We group regions of $5 \times 5 \times 1$ voxels and considered overlapping groups “shifted" by 2 voxels in the first 2 dimensions.The irregular group size compensates for voxels being larger and scanner coverage being smaller

in the z-dimension (only 8 slices relative to 64 in the x- and y-dimensions).

Figure 29 shows the prediction error (misclassification rate) of each classifier for every individual subject. SOSlasso shows the smallest error. The substantial gains over lasso indicate that the algorithm is successfully leveraging cross-subject consistency in the location of the informative features, allowing the model to avoid over-fitting individual subject data. We also note that the SOSlasso classifier, despite being trained without any voxel pre-selection, averaging, or alginment, performed comparably to the best-performing classifier reported by [88], which was trained on features average over 7 expert pre-selected ROIs

To assess how well the clusters selected by SOSlasso align with the anatomical regions thought a-priori to be involved in sentence and picture representation, we calculated the proportion of selected voxels falling within the 7 ROIs identified by [88] as relevant to the classification task (Table 6). For SOSlasso an average of 61.9% of identified voxels fell within these ROIs, significantly more than for lasso, group lasso (with or without overlap) and the elastic net. The overlapping group lasso, despite returning a very large number or predictors, hardly overlaps with the regions of interest to cognitive neuroscientists. The lasso and the elastic net make use of the fact that a separate classifier can be trained for each subject, but even in this case, the overlap with the regions of interest is low. The group lasso also fares badly in this regard, since the same voxels are forced to be selected across individuals, and this means that the regions of interest which will be misaligned across subjects will not in general be selected for each subject. All these drawbacks are circumvented by the SOSlasso. This shows that even without expert knowledge about the relevant regions of interest, our method partially succeeds in isolating the voxels that play a part in the classification task.

We make the following observations from Figure 29 and Figure 30

- The overlapping group lasso [37] is ill suited for this problem. This is natural, since

Method	Avg. Overlap with ROI %
OGlasso	27.18
ENet	43.46
Lasso	41.51
Glasso	47.43
SOSlasso	61.90

Table 6: Mean Sparsity levels of the methods considered, and the average overlap with the precomputed ROIs in [88]

the premise is that the brains of different subjects can only be crudely aligned, and the overlapping group lasso will force the same voxel to be selected across all individuals. It will also force all the voxels in a group to be selected, which is again undesirable from our perspective. This leads to a high number of voxels selected, and a high error.

- The elastic net [95] treats each subject independently, and hence does not leverage the inter-subject similarity that we know exists across brains. The fact that all correlated voxels are also picked, coupled with a highly noisy signal means that a large number of voxels are selected, and this not only makes the result hard to interpret, but also leads to a large generalization error.

- The lasso [83] is similar to the elastic net in that it does not leverage the inter subject similarities. At the same time, it enforces sparsity in the solutions, and hence a fewer number of voxels are selected across individuals. It allows any task correlated voxel to be selected, regardless of its spatial location, and that leads to a highly distributed sparsity pattern (Figure 30a). It leads to a higher cross-validation error, indicating that the ungrouped voxels are inferior predictors. Like the elastic net, this leads to a poor generalization error (Figure 29). The distributed sparsity pattern, low overlap with pre-determined Regions of Interest, and the high error on the hold out set is what we believe

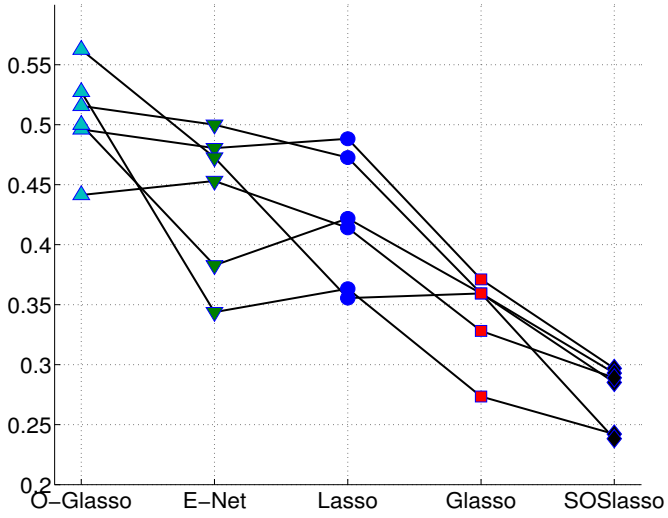


Figure 29: Misclassification error on a hold out set for different methods, on a per subject basis. Each solid line connects the different errors obtained for a particular subject in the dataset.

makes the lasso a suboptimal procedure to use.

- The group lasso [49] groups a single voxel across individuals. This allows for taking into account the similarities between subjects, but not the minor differences across subjects. Like the overlapping group lasso, if a voxel is selected for one person, the same voxel is forced to be selected for all people. This means, if a voxel encodes picture or sentence in a particular subject, then the same voxel is forced to be selected across subjects, and can arbitrarily encode picture or sentence. This gives rise to a purple haze in Figure 30b, and makes the result hard to interpret. The purple haze manifests itself due to the large number of ambiguous voxels in Figure 30d.

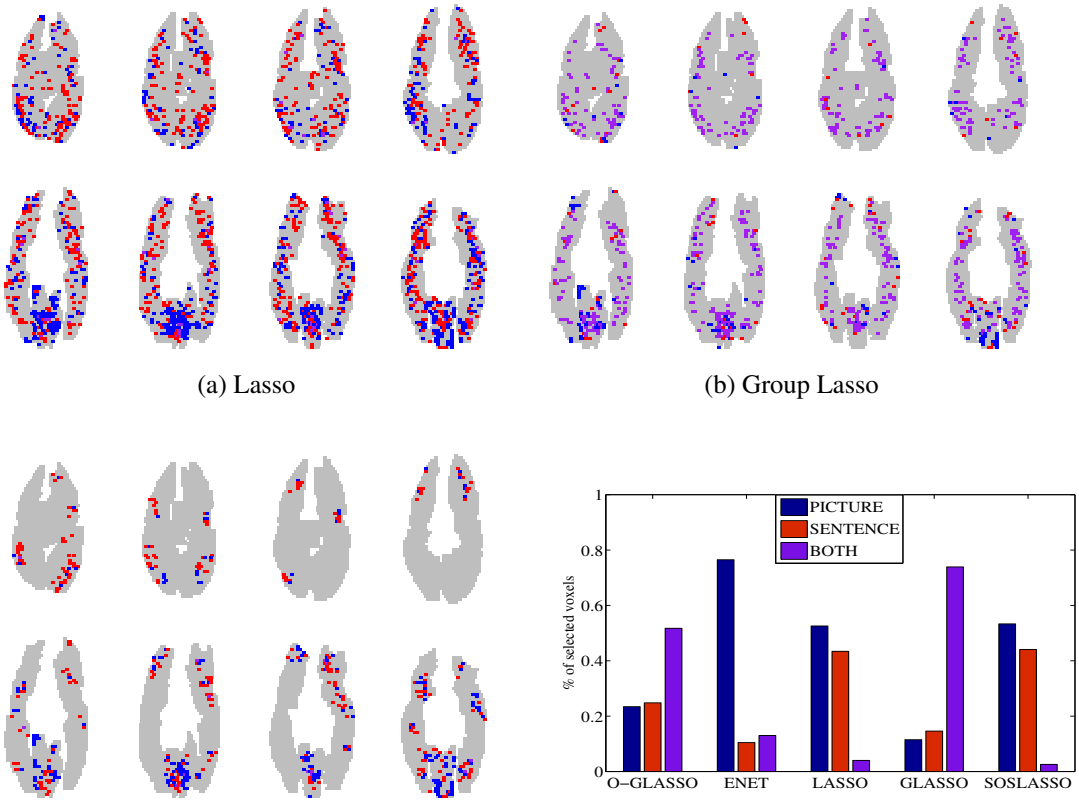
- Finally, the SOSlasso as we have argued helps in accounting for both the similarities and the differences across subjects. This leads to the learning of a code that is at the same

time very sparse and hence interpretable, and leads to an error on the test set that is the best among the different methods considered. The SOSlasso (Figure 30c) overcomes the drawbacks of lasso and Glasso by allowing different voxels to be selected per group. This gives rise to a spatially clustered sparsity pattern, while at the same time selecting a negligible amount of voxels that encode both picture and sentences (Figure 30d). Also, the resulting sparsity pattern has a larger overlap with the ROI's than other methods considered.

4.2 The Sparse Overlapping Sets lasso for Gene Selection

As explained in the introduction, another motivating application for the SOSlasso arises in computational biology, where one needs to predict whether a particular breast cancer tumor will lead to metastasis or not, from gene expression profiles. Genes are typically organized into pathways, with genes in a single pathway being correlated. If a particular gene is found to be relevant for prediction of a disease, then it is likely that she correlated genes will also be relevant. Moreover, these pathways overlap with each other. However, pathways are typically 100's of genes long, and this makes the overlapping group lasso ill-suited for this problem. Indeed, if a cartain pathway is found to be active, then all the genes in that pathway will be activated, making the solution hard to interpret. It is reasonable to assume that if a certain gene is relevant, then *some* correlated genes will be relevant, and that makes the SOSlasso a perfect candidate so solve this problem.

We used the breast cancer dataset compiled by [86] and grouped the genes into pathways as in [80]. To make the dataset balanced, we perform a 3-way replication of one of the classes as in [37], and also restrict our analysis to genes that are atleast in one pathway. Again as



(a) Lasso

(b) Group Lasso

(c) SOS Lasso

(d) Voxel Encodings (%)

Figure 30: [Best seen in color]. Aggregated sparsity patterns across subjects per brain slice. All the voxels selected across subjects in each slice are colored in red, blue or purple. Red indicates voxels that exhibit a picture response in at least one subject and never exhibit a sentence response. Blue indicates the opposite.Purple indicates voxel that exhibited a a picture response in at least one subject and a sentence response in at least one more subject. (d) shows the percentage of selected voxels that encode picture, sentence or both.

in [37], we ensure that all the replicates are in the same fold for cross validation. We do not perform any preprocessing of the data, other than the replication to balance the dataset. We compared our method to the standard lasso, and the overlapping group lasso. The standard group lasso [91] is ill-suited for this experiment, since the groups overlap and the sparsity pattern we expect is a union of groups, and it has been shown that the group lasso method will not recover the signal in such cases.

Method	**Misclassification Rate**
lasso	0.42
OGlasso [37]	0.39
SOSlasso	0.33

Table 7: Misclassification Rate on the test set for the different methods considered. The SOSlasso obtained better error rates as compared to the other methods.

We trained a model using 4-fold cross validation on 80% of the data, and used the remaining 20% as a final test set. Table 7 shows the results obtained. We see that the SOSlasso penalty leads to lower classification errors as compared to the lasso or the latent group lasso. The errors reported are the ones obtained on the final (held out) test set.

We see that the SOSlasso leads to lower misclassification errors as compared to the lasso and the overlapping group lasso. Note that we did not compare with the standard group lasso in this case, since it is known that the groups overlap.

4.3 Convex Approaches to Model Wavelet Sparsity Patterns

We now deviate from the SOSlasso, and consider a novel application of the overlapping group lasso in compressive imaging, that is to recover an image from a small number of random measurements. Here “small" is used relative to the ambient dimension of the image.

We let $\boldsymbol{x}$ denote the inverse Discrete Wavelet Transform (DWT) coefficients of an image. Let $\boldsymbol{\Phi}$ be a sensing matrix, and suppose we observe noisy linear measurements of the form

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x} + \eta \qquad \eta \sim \mathcal{N}(0, \sigma^2 \boldsymbol{I})$$

where η is Additive White Gaussian Noise (AWGN).

The standard lasso [83] formulation is given by

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}\|^2 + \lambda\|\boldsymbol{x}\|_1 \tag{4.1}$$

The ℓ_1 norm acts as a surrogate for the sparsity of the signal. The lasso aims to recover a signal that is sparse, by setting most coefficients of $\boldsymbol{x}$ to be zero. For the exact recovery case, the lasso problem is equivalent to the Basis Pursuit [16]

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \|\boldsymbol{x}\|_1 \quad \text{s.t. } \boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x} \tag{4.2}$$

The lasso penalty reflects the fact that the wavelet coefficients are approximately sparse, but in reality not all patterns of sparsity are equally plausible/probable. For example, Fig (31b) shows the DWT coefficients of the barbara image, and Fig. (31c) shows the same coefficients, but randomly scrambled. Clearly, the ℓ_1 norm of both sets of coefficients will be the same. This shows that the lasso penalty in itself is invariant to any structure present in the sparse coefficients.

To model this structure that is inherently present between wavelet transform coefficients of images, [19, 26, 73] propose making use of graphical models such as Hidden Markov Trees (HMT's). HMT's, while providing good performance in image denoising applications (where

(a) Original image

(b) 3 -stage DWT of the barbara image

(c) Coefficients of the DWT of the Barbara image, randomized

Figure 31: The ℓ_1 norms of both (b) and (c) are exactly equal, since they do not take structure into account

$\mathbf{\Phi} = \boldsymbol{I}$) in (4.1), cannot provide acceptable reconstruction for other, more general inverse problems. This is because the presence of a (non identity) sensing matrix $\mathbf{\Phi}$ (randomly) mixes up the coefficients for every measurement $\boldsymbol{y}_i$ obtained.

To overcome this mixing between the coefficients, many alternatives have been proposed. [75] propose using a version of loopy belief propagation to solve the recovery problem. The authors in [5, 25] generalize the notion of restricted isometry properties to signals that lie in unions of subspaces, and use a modified version of CoSAMP [58] to solve the inverse problem. Greedy and/or suboptimal iterative reconstruction schemes are used in [26, 46]. Finally, the authors in [77] propose modeling the coefficients using an HMT, and using the Approximate Message Passing algorithm [23] to solve the compressed sensing problem.

All the methods mentioned above sacrifice the recovery guarantees and the easy analysis that convex optimization algorithms provide, for the sake of modeling the dependencies between DWT coefficients (an exception being [5] that provide guarantees for the greedy

scheme developed). This motivates our problem: can we on the one hand model the dependencies among wavelet transform coefficients, while at the same time propose to solve a convex optimization problem similar to (4.1)?

To this end, we model the parent-child coefficients into groups. Parent-child pairs of wavelet transform coefficients across scales and at similar locations tend to be simultaneously high or low. Hence, we can take advantage of this dependency and use group lasso methods to recover the image. Fig 32 shows a representative example.

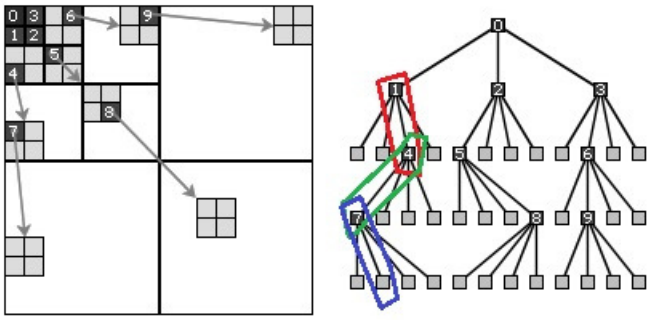


Figure 32: Quadtree corresponding to the 2-d DWT. At each scale, parent coefficients can be grouped with child coefficients.

Note that modeling the coefficients into parent child pairs is more advantageous than looking for a rooted tree by forming groups along paths of the tree. It has been established that the coefficients of an image do not form a rooted tree.

As an illustrative example, consider the standard “blocks" signal, and its Haar DWT coefficients (Figure 33). It is possible that at a certain level j and location k, the sum of the signal values corresponding to the positive part of the Haar basis vector cancels with that of the negative part, leading to a small (or zero) wavelet coefficient. However, as we move to finer scales, the variations in the signal may mimic the variations in the Haar wavelet support,

resulting in large values of the corresponding coefficients, as can be seen in Fig. 34.

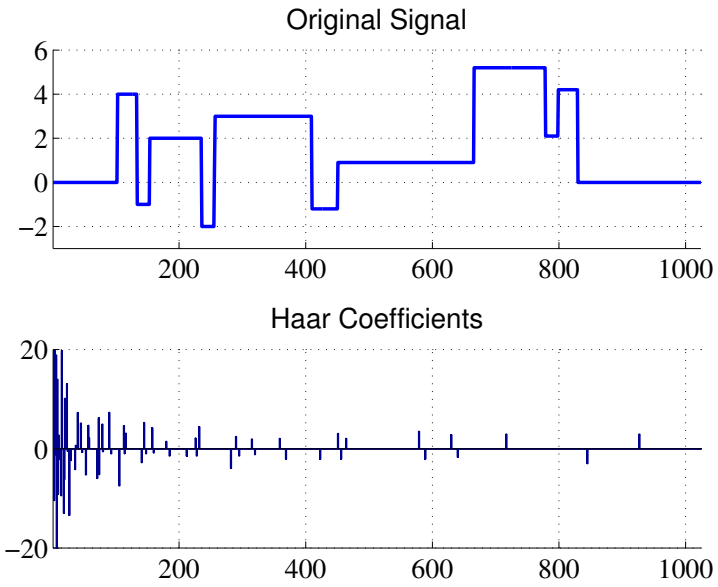


Figure 33: 'Blocks' and its Haar DWT

Hence, there do exist cases where a parent coefficient is inactive(zero), and it's child is active (non zero). The fact that we only group pairs of nodes in the tree allows for a group higher up in the tree to be zeroed out, while still having non zero coefficients from groups beneath it. This is another advantage of our method over other methods that enforce a tree-like set of coefficients.

We wish to recover the non zero coefficients lying on the wavelet tree shown in Fig 32. When coefficients are modeled into groups, one can use the group lasso [91] to recover the coefficients

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}\|^2 + \lambda \sum_{i=1}^{M} \|\boldsymbol{x}_{G_i}\| \tag{4.3}$$

where $\boldsymbol{x}_{G_i}$ is the vector $\boldsymbol{x}$ whose coefficients not indexed by group G_i are set to zero. The group lasso as shown in (4.3) suffers from a drawback however. We showed in Chapter 2 that

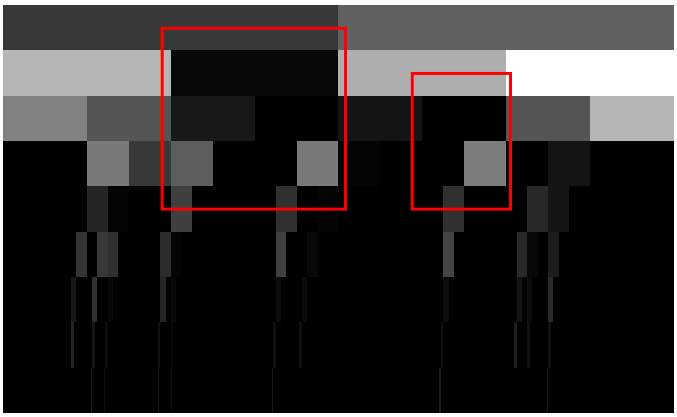


Figure 34: (Best seen in color) Haar coefficeints in Figure 33 arranged in a tree. Darker regions correspond to smaller magnitude coefficients. We see that wavelet coefficients can be small (or zero) and still have non zero children, as denoted by the red rectangles.

the sparsity pattern recovered by the group lasso can be expressed as a complement of a union of groups. One look at Figure 32 tells us that we are interested in the recovery of sparsity patterns that can be expressed as a union of (overlapping) groups. To this end, the authors in [37] propose the latent group lasso [63, 64]

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}\|^2 + \lambda \Omega^{\mathcal{G}}_{overlap}(\boldsymbol{x}) \tag{4.4}$$

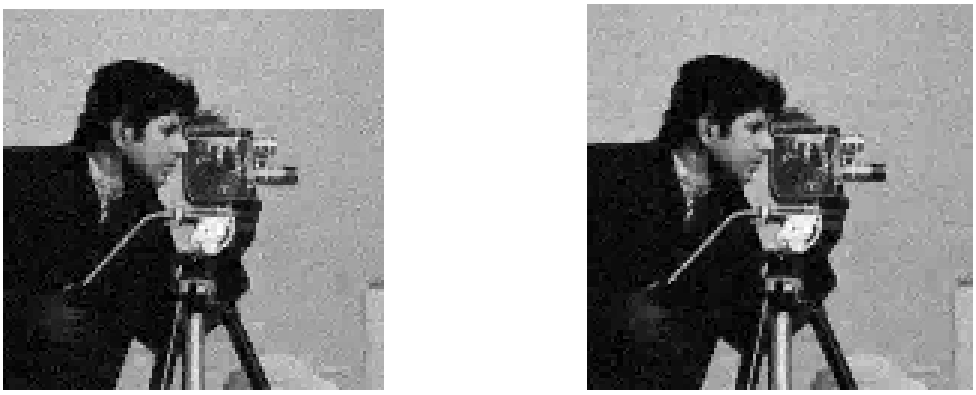
where $\Omega^{\mathcal{G}}_{overlap}(\boldsymbol{x})$ is the latent group lasso norm. defined by

$$\Omega^{\mathcal{G}}_{overlap}(\boldsymbol{x}) = \min_{\mathcal{V}=[\boldsymbol{v}_{G_i}]} \sum_{i=1}^{M} \|\boldsymbol{v}_{G_i}\| \quad \text{s.t. } \boldsymbol{x} = \sum_{i=1}^{M} \boldsymbol{v}_{G_i} \tag{4.5}$$

To solve the problem, we make use of SpaRSA [90] after performing replication, which is explained in Chapter 2.

We considered a 128×128 section of the cameraman image (normalized to lie in $[0, 1]$), and obtained 6000 noisy ($\sigma = 0.05$) iid Gaussian measurements from it. We compare our

recovery to the standard Lasso. In Figure 35, we see that the recovery using the group lasso is better in terms of PSNR.



(a) lasso PSNR = 25.73 dB (b) Glasso PSNR = 28.29 dB

Figure 35: Reconstruction of a section of the cameraman image using lasso and group lasso

We performed reconstruction experiments on natural images from the “background" set of 550 images from the Caltech image database (http://www.vision.caltech.edu/html-files/archive.html). We used the first 350 images in the dataset for denoising experiments, and the rest for deconvolution experiments. For denoising, every image was resized to size 64 X 64, normalized to have range $[0, 1]$ and vectorized to length 4096. 800 Gaussian samples were used per image to reconstruct it, after corrupting it with AWGN of variance 0.5. For the deconvolution case, the image was blurred with a Gaussian kernel of variance 0.5. Table 8 shows the average squared reconstruction error (per pixel), over all the 350 images considered for denoising, and 200 images considered for deconvolution. The regularization parameters were learned over a grid, ranging from 10^{-2} to 10^{3} for respective λs

Figure 36 shows the results we obtain as a function of the noise standard deviation. We consider piecewise constant signals of length 1024, having 7 jumps. The location of the jumps

Method	MSE (denoise)	MSE (deconv)
lasso	0.0172	0.0194
OGLR	0.0107	0.0120

Table 8: Denoising and deconvolution performance on the “background" dataset. Column 2 displays the mean reconstruction error for the denoising experiments, while column 3 does the same for deconvolution.

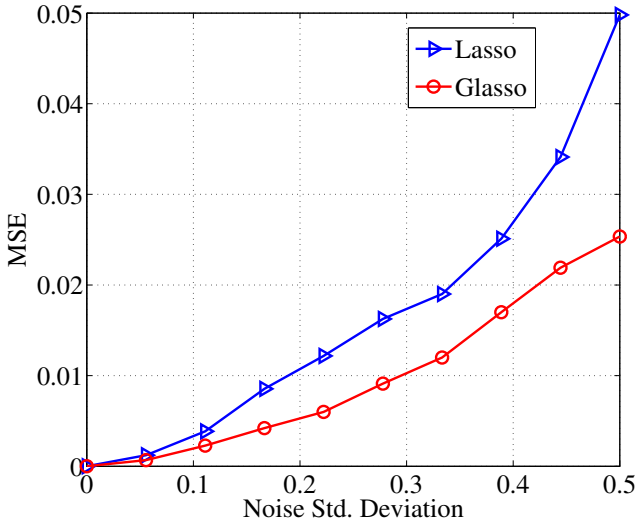


Figure 36: Comparison of the two methods in the presence of noise.

is chosen at random, and the magnitude of each “piece" is chosen uniformly between $[-1, 1]$. We take 256 measurements for both the lasso and Glasso. From the figure, it is clear that by modeling the wavelet coefficients into parent-child pairs, we can better reconstruct signals in the presence of noise. The results are averaged over 1000 randomly generated signals.

We consider the “peppers" image, sized to 128 X 128. We vectorize the image to length 16384, and take 5000 Gaussian measurements. We considered this image so as to compare our results with those in [5]. We see that (Figure 37), we can recover the signal accurately, with very few measurements. The figure also shows the results obtained by [5], where the authors also take 5000 measurements.

(a) Peppers (b) CoSaMP (MSE = 22.8) (c) Model-Based (MSE = 11.1) (d) OG Lasso (MSE = 7.8)

Figure 37: Performance on the peppers image

Finally, we use CoGEnT to perform compressive image recovery. We consider the popular Shepp-Logan phantom image, resized to 64×64. We vectorize the image and obtain 2000 i.i.d Gaussian measurements, and look to reconstruct the image. Figure 38 shows that CoGEnT does help in solving structured compressed sensing problems, without blowing up the problem dimension. Note that the resulting dimension of the problem after replication is much larger than the true dimension.

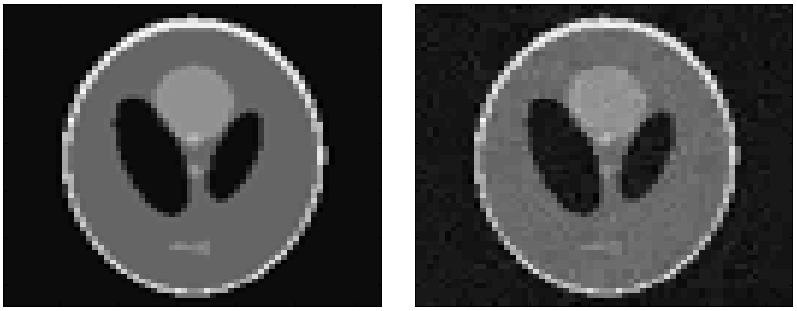


Figure 38: Compressive image recovery using CoGEnT . Left: Original image. Right: recovered image, PSNR = 32.7 dB. Note that we deal with 4096 dimensional signals in this experiment. If we use the strategy in [67], the corresponding dimension after replication would be 15680.

4.4 Sensing Matrix Design for Compressive Imaging

In this section, we extend the method considered in the previous section, and incorporate prior knowledge about variable grouping into the design of the sensing matrix itself. We aim to show that by redesigning the (standard) i.i.d. Gaussian sensing matrix to reflect the intra group dependencies, significant improvements in image reconstruction are possible.

Consider the standard compressive sensing framework of images, the same as that considered in the previous section:

$$\begin{aligned} \boldsymbol{y} &= A\theta \ + \eta \\ &= AW^{-1}\boldsymbol{x} + \eta \\ &= \boldsymbol{\Phi}\boldsymbol{x} + \eta \end{aligned}$$

where $\boldsymbol{y} \in \mathbb{R}^m$ is a vector of measurements, $A \in \mathbb{R}^{m \times n}$ is the sensing matrix, with $m < n$. W is the DWT matrix, and $\theta \in \mathbb{R}^n$ is the image (vectorized). $\boldsymbol{x}$ is the DWT coefficients of the image, which is known to be (approximately) sparse. $\eta \in \mathbb{R}^m$ is an i.i.d. Gaussian noise vector of zero mean and unit variance.

As in the previous section, we assume that the wavelet coefficients x can be grouped into parent-child pairs (Figure 32), and solve the overlapping (latent) group lasso program:

$$\widehat{\boldsymbol{x}}_{Glasso} = \quad \arg\min_{\boldsymbol{x}} \ \frac{1}{2}\|\boldsymbol{y} - A_{iid}W^{-1}\boldsymbol{x}\|^2 + \lambda_g \Omega^{\mathcal{G}}_{overlap}(\boldsymbol{x}) \tag{4.6}$$

where $\Omega^{\mathcal{G}}_{overlap}(\boldsymbol{x})$ is the latent group lasso penalty [63]. For a set of groups $G_1, G_2, \ldots, G_M$,

the latent group lasso penalty is defined by

$$\Omega^{\mathcal{G}}_{overlap}(\boldsymbol{x}) = \inf_{\sum_i \boldsymbol{v}_i = \boldsymbol{x}} \sum_{i=1}^{M} \|\boldsymbol{v}_i\|$$

where $\boldsymbol{v}_i \in \mathbb{R}^n$ has support restricted to the indices in group G_i. A_{iid} is the standard Gaussian i.i.d. sensing matrix used for compressed sensing. The latent group lasso penalty has the property that the pattern of non zeros (coefficients) recovered by solving (4.6) can be expressed as a union of groups.

We continue to build on the same model for $\boldsymbol{x}$ as introduced in the previous section: Assume that the parent child groups are given by $\mathcal{G} = \{G_i\}_{i=1}^{M}$. We assume a mixture model, where a group can be active with probability p. Since the signal is group-sparse, p is small ($p << 1$). If a group is active, we assume that the coefficients arise from a distribution with covariance matrix Σ. The matrix Σ encodes the intra group dependencies between variables. Since we are considering only parent-child pairs, $\Sigma \in \mathbb{R}^{2\times 2}$, and the off-diagonal elements will represent the cross-correlation between the parent and child

Our goal in this section is to learn the covariance matrix coresponding to the parent child pair from a training set of images, and use this as prior information to design new sensing matrices that are better matched to the image structure [2]. In doing so, the sensing energy is better aligned to the image structure, and consequently active groups can be reliably identified using fewer measurements than needed with the conventional sensing matrix composed of i.i.d. zero-mean Gaussian variables.

An important point to note is that since the covariance matrix that we aim to learn is only 2×2, we do not need as many samples as one might need to accurately learn the entire $n \times n$

[2]Note that an alternative is to use the universal Hidden Markov Model [73] where the covariance matrix can directly be inferred using the transition probabilities between parent and child states

covariance matrix corresponding to the images.

4.4.1 Measurement Matrix Design

A group sparse signal can be written as a sum of signals whose support is restricted to be the indices corresponding to a single group. For example, the signal s comprising of three groups in Figure 39 can be decomposed using latent variables [63] into signals s_1, s_2 and s_3

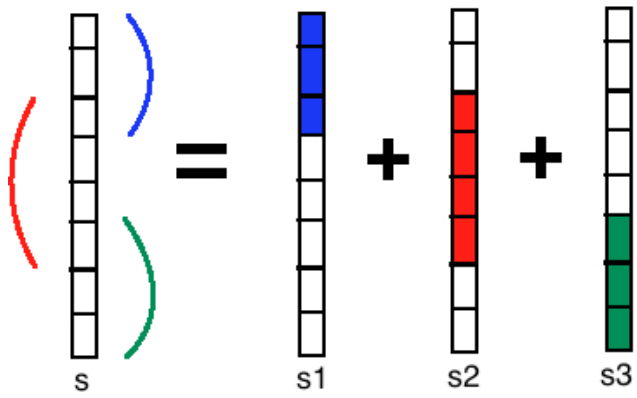


Figure 39: Decomposition of the group sparse signal in latent group lasso. The curved lines in the LHS represent groups of coefficients. (best seen in color)

The latent groups can be active/inactive independent of each other, and the final sparse signal is the sum (normalized) of the decomposition. It can then be seen that, if the groups G_i have covariance Σ_i, then the final (sum) vector will have a covariance matrix that can be decomposed as in Figure 40, where C is the matrix after combining the individual covariance matrices Σ_i (shown shaded in Fig 40).

Based on this insight, we form measurement vectors (the rows of the measurement matrix): Assume we are given the covariance matrices $\Sigma_i = \Sigma$. We assume all the covariance matrices are the same, for simplicity. We note later in the chapter that one can assume the covariance

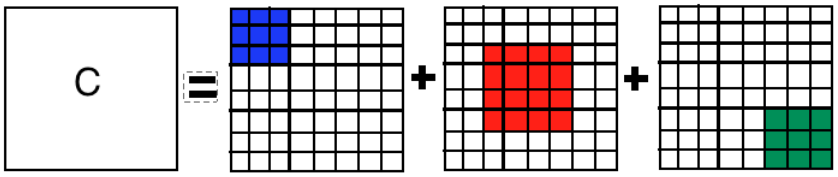


Figure 40: Decomposition of the covariance matrix of the signal in Figure 39. The shaded parts in the RHS if the figure correspond to covariance matrices of the individual active groups, Σ_i, $i = 1, 2, 3$.

matrices to be different across scales, and still apply our method. Let $a^i \in \mathbb{R}^n$ denote the i-th row of the sensing matrix under construction. We start with $a^i = 0$. Letting a_{G_i} be the sub-vector of a^i indexed by group G_i, we recursively perform the following operation $\forall G_i \in \mathcal{G}$:

$$\boldsymbol{a}_{G_i} = \boldsymbol{a}_{G_i} + \boldsymbol{v}_{G_i} \ , \tag{4.7}$$

where

$$\boldsymbol{v}_{G_i} \sim \mathcal{N}(0, \boldsymbol{\Sigma}) \ .$$

This procedure yields $\boldsymbol{a}^i \sim \mathcal{N}(0, \boldsymbol{C})$, where $\boldsymbol{C}$ is the covariance matrix obtained as a result of the composition of the vectors, as in Figure 40. The measurement matrix is generated by repeating this procedure m times. We then normalize the columns to have unit norm, and call this matrix A_{avg}, distinguishing it from the standard Gaussian sensing matrix used for compressed sensing, which we denote by A_{iid}. The columns of A_{iid} are also normalized, so that both A_{avg} and A_{iid} have Frobenius norm n. This puts the two sensing matrices on equal footing, as far as SNR is concerned. Hence, we now solve the group lasso as in (4.6), but with A_{avg}:

$$\widehat{\boldsymbol{x}}_{CGlasso} = \quad \arg\min_{\boldsymbol{x}} \ \frac{1}{2}\|\boldsymbol{y} - A_{avg}W^{-1}\boldsymbol{x}\|^2 + \lambda_a \Omega^{\mathcal{G}}_{overlap}(\boldsymbol{x}) \tag{4.8}$$

The intuition for this approach arises from work on correlated Gaussian designs, as in [69]. In [69], it was shown that Gaussian matrices with non *iid* columns also obey the restricted eigenvalue conditions [8] needed for exact recovery in the noiseless setting, and robust recovery in the noisy case. In our case, the sparse vector to be reconstructed itself has correlated entries. To take into account this correlation, we allow the columns in the measurement matrix (called predictor variables in sparse linear regression settings) to be correlated. Hence, selection of any variable will automatically force the selection of a correlated variable, when we use the latent group lasso.

It is well known that to accurately capture a signal, one should measure along the direction of the signal. The matched filter is a typical example that makes use of this principle. By generating sensing vectors from (roughly) the same distribution as the data itself, we ensure that the measurement is correlated with the signal itself, facilitating better recovery. Note that in this case, the “data" corresponds to coefficients from a single group.

We refer to the method introduced in the previous section as Glasso and the method we developed by CGlasso, the ‘C’ indicating the use of the covariance matrix in our design. For details on how the groups are designed, and the Glasso method, we refer the reader to the previous section. Again, we solve the latent group lasso problem by the replication strategy elaborated in [37]. We use SpaRSA [90] to solve the optimization problems.

To show the efficacy of our method, we first consider a toy signal, with a known covariance matrix. Note here that the covariance matrix refers to the 2×2 matrix corresponding to the parent child pairs on the DWT tree. We assume a signal consisting of 100 non-overlapping groups of size 10 each, of which 10 are active, and consider 250 measurements. The signals have active groups whose coefficients are generated from a zero mean Gaussian with covariance matrix $U^T U$, where $U \in \mathbb{R}^{10\times 10}$ is an orthonormal basis for a 10×10 random Gaussian

matrix. It can be seen from Figure 41 that CGlasso outperforms Glasso

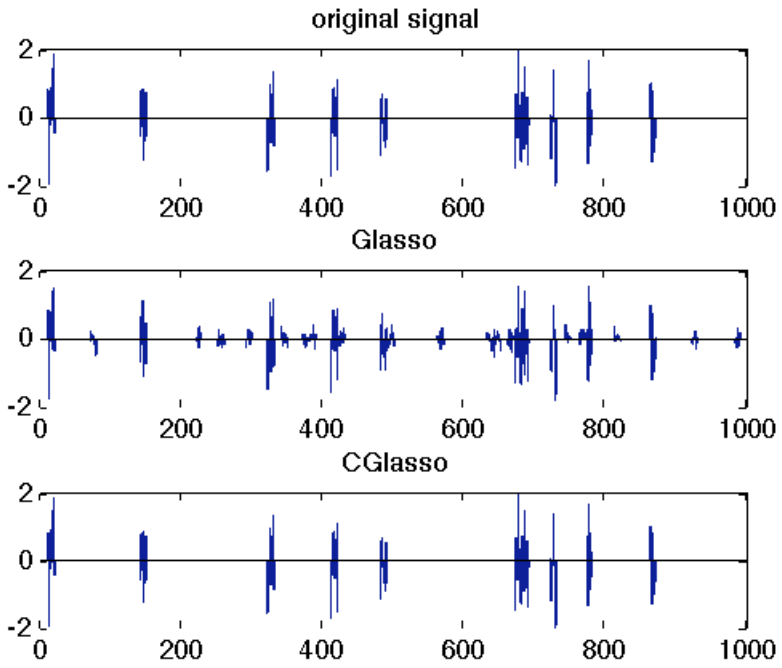


Figure 41: Comparison of the two methods. It can be seen the CGlasso recovers the signal exactly, while Glasso makes some errors, for the given number of measurements.

We also compared the overlapping group lasso to our new method, with noisy measurements. Table 9 demonstrates that CGlasso outperforms Glasso under this scenario as well. We again considered the same toy signals as the ones explained with reference to Figure 41. The signals were of length 1000, and we considered 200 measurements. The results are averaged over 100 tests. Note that once the number of measurements exceeds a certain bound, Glasso also gives near exact recovery.

Noise Std. Dev	**MSE CGlasso**	**MSE Glasso**
0	$\approx 10^{-31}$	0.0200
0.02	0.0001	0.0225
0.04	0.0003	0.0294
0.06	0.0006	0.0355
0.08	0.0011	0.0497
0.1	0.0019	0.0556

Table 9: Comparison of the two methods under noisy measurements.

Chapter 5

Future Directions and Conclusions

5.1 Future Directions

In this thesis, we focussed on the problem of structured sparse pattern recovery, and derived novel theoretical guarantees, regularization functions and algorithms. This thesis added to the rapidly growing literature in the field of structured pattern recovery, and below we outline some open problems that arise as a result of this thesis. Some of the works mentioned below are ongoing, while some are left as future work.

The SOSlasso framework allows one to recover patterns that can be expressed as a combination of overlapping group sparse and sparse patterns. Another way of looking at this problem, is that of recovery of a hierarchical sparsity pattern, where there exists overlapping groups in one level of the hierarchy, and groups of size 1 in the level below it (Figure 42).

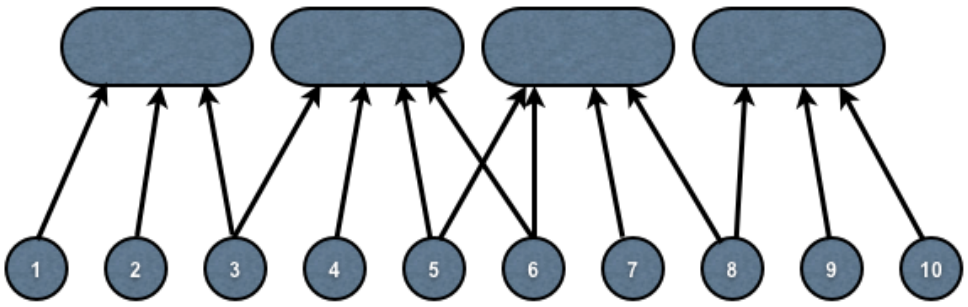


Figure 42: Setup for the SOSlasso. Variables 1-3 are in one group, variables 3-6 are in another group, variables 5-8 are in the third group and the variables 8-10 are in the final group

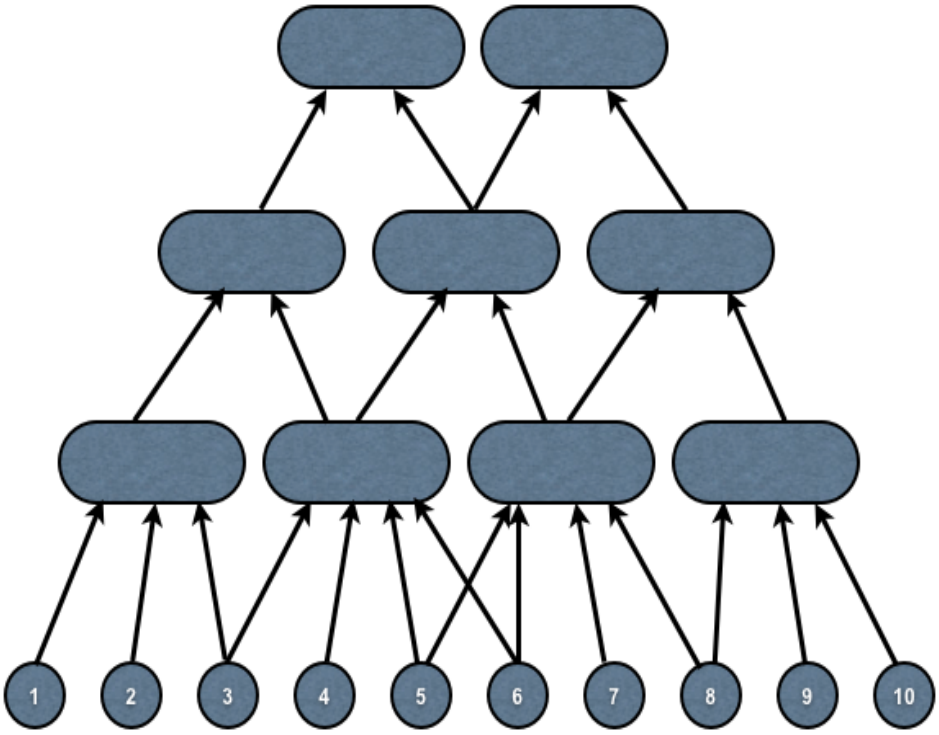


Figure 43: Hierarchical overlapping groups. We start with the same configuration as for the SOSlasso, but the groups themselves can be grouped into sets in higher levels

In some applications, it makes sense to consider a similar situation, but with more levels in the hierarchy. Specifically, consider the case of learning semantic labels from images (or documents). The lowermost level might correspond to objects in the image (like net, table etc.). These objects can be grouped into overlapping sets, of household items, sport equipment, etc. Furthermore, these sets themselves can be grouped into overlapping sets, representing higher order semantic information, and so on. This gives rise to a structured sparsity pattern comprising of multiple levels of overlapping groups, as shown in Figure 43 as an extension of the SOSlasso. Figure 44 shows a more concrete example

Future work involves investigating this sparsity pattern, and its application to various problems of interest. Like in the case of images, we can consider learning contents of documents

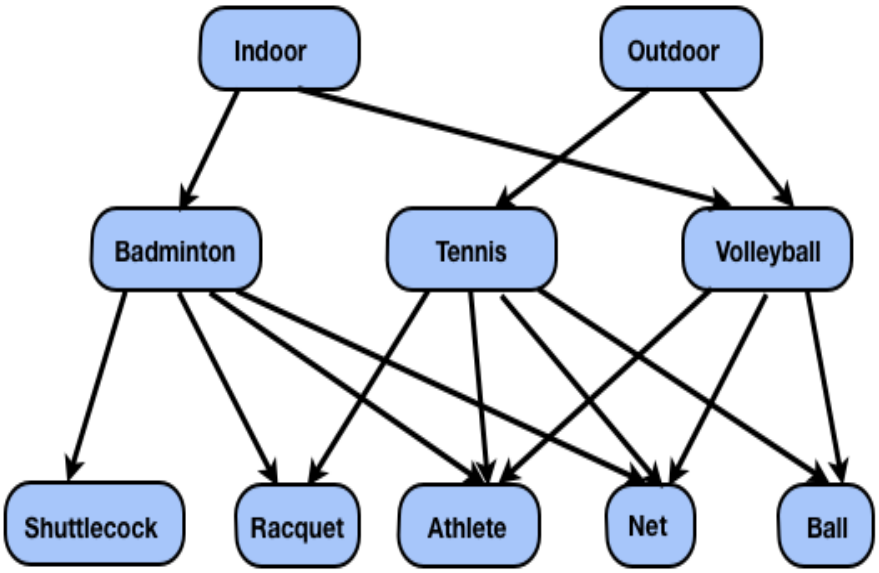


Figure 44: Hierarchical overlapping groups for learning higher order information from objects in scenes

as well, from a bag-of-words feature representation.

When the groups overlap a lot, the SOSlasso with replication suffers from the same drawback as the group lasso with replication: it leaves a large memory footprint. CoGEnT alleviates this problem for the overlapping group lasso, but the SOSlasso does not admit a simple atomic norm based representation. At the same time, since we have shown that CoGEnT can be used even with approximately optimal steps, it is interesting to see if a strategy can be devised for the SOSlasso as well.

Another possible direction for future research is to succinctly characterize the atomic set for the SOSlasso. A very simple characterization of the atomic set can be written as:

$$A_G = \{\boldsymbol{a} : \text{supp}(\boldsymbol{a}) \subset G, \;\; \|\boldsymbol{a}\|_2 + \|\boldsymbol{a}\|_1 = 1\}$$

$$\mathcal{A} = \bigcup_{G \in \mathcal{G}} A_G$$

It will be an interesting exercise to study in detail this atomic set, and the apply CoGEnT and/ or there techniques to solve and characterize the SOSlasso problem.

From a cognitive neuroscience point of view, future work involves grouping the voxels in more intelligent ways. Our method to group spatially co-located voxels yields results that are significantly better than traditional lasso-based methods, but it remains to be seen whether there are better motivated ways to group them. For example, one might consider grouping voxels based on functional connectivities, or take into account the geodesic distance on the brain surface. One can also look to apply an SOSlasso type regularization in the graphical lasso setting, for inferring functional connectivity.

For CoGEnT , we aim to handle general convex loss functions. The analysis does not rely on the fact that the loss function is the least squares loss, and hence it should be fairly straightforward to obtain results for other loss functions as well. Note that for the linear convergence rate, we do make use of the fact that the loss function is the least squares loss. Along similar lines, our experiments suggest that CoGEnT not only yields sparser solutions, but converges much faster than standard conditional gradient. It will be interesting to see whether this increased convergence rate amounts to constant factor improvements in the theoretical rate we derived, or if we can get rates much faster than $\frac{1}{T}$.

Also, it will be interesting to see what other applications CoGEnT can be applied to, especially those cases where no simple method exists currently. We also intend to extend the method to online and/or distributed settings.

5.2 Conclusion

In this thesis, we developed theoretical results for structured sparse signal recovery. We introduced a method called the Sparse Overlapping Sets Lasso, and showed that it generalizes the lasso, group lasso and the group lasso with overlapping groups to incorporate sparsity patterns that can be expressed as a union of sparsely populated groups. We derived sample complexity bounds for the SOSlasso under classification and regression settings. We also derived sample complexity bounds for the group lasso with overlapping groups for linear regression.

We also developed CoGEnT , a method to perform very general high dimensional structured recovery. This method not only generalizes the standard greedy methods for recovery of sparse vectors or low rank matrices, but in some cases achieves state of the art performance, while in others is the first of it's kind to solve certain problems.

Lastly, this thesis demonstrated the use of structured sparse pattern recovery methods to problems in fMRI, computational biology and image processing. We showed that by 1) understanding that there is structure within the coefficients of the desired signal of interest, and 2) developing a scheme that takes advantage of this structure, we can achieve state of the art results in some applications, while in others, we believe we have opened the doors for further development of these methods and achieving much better results.

5.3 Full List of Publications

1. *Logistic Regression with Structured Sparsity*: Nikhil Rao, Robert Nowak, Chris Cox and Timothy Rogers, arXiv:1402.4512, 2014 (In preparation for submission to the Journal of Machine Learning Research)

2. *A Forward-backward Algorithm for Atomic Norm Regularization*: Nikhil Rao, Parikshit Shah and Stephen Wright, arXiv:1404.5692, 2014 (submitted to IEEE Trans. Signal Processing)

3. *Forward-Backward Greedy Algorithms for Signal Demixing*: Nikhil Rao, Parikshit Shah and Stephen Wright, Asilomar Conference on Signals, Systems and Computers (Invited paper), 2014

4. *Sparse Overlapping Sets Lasso for Multitask Learning and fMRI Data Analysis*: Nikhil Rao, Christopher Cox, Robert Nowak and Timothy Rogers, Neural Information Processing Systems 2013 (Spotlight Presentation)

5. *Conditional Gradient with Enhancement and Truncation for Atomic Norm Regularization*: Nikhil Rao, Parikshit Shah and Stephen Wright, NIPS workshop on Greedy Algorithms 2013

6. *A Greedy Forward Backward Method for Atomic Norm Constrained Minimization*: Nikhil Rao, Parikshit Shah, Stephen Wright and Robert Nowak, IEEE International Conference on Acoustics, Speech and Signal Processing, 2013

7. *Adaptive Sensing with Structured Sparsity*: Nikhil Rao, Gongguo Tang and Robert Nowak, IEEE International Conference on Acoustics, Speech and Signal Processing, 2013

8. *Knowledge Enhanced Measurements for Estimating Sparse Signals from Clutter*: Workshop on Signal Processing with Adaptive Structured Sparse Representations (SPARS) 2013, Lausanne, Switzerland

9. *Adaptive Sensing on Markov Trees* : Workshop on Signal Processing with Adaptive Structured Sparse Representations (SPARS) 2013, Lausanne, Switzerland

10. *Correlated Gaussian Designs for Compressive Imaging* : Nikhil Rao and Robert Nowak, IEEE International Conference on Image Processing, 2012

11. *Universal Measurement Bounds for Structured Sparse Signal Recovery* : Nikhil Rao, Benjamin Recht and Robert Nowak, Journal of Machine Learning Research (proc. Artificial Intelligence and Statistics), 2012

12. *A Clustering Approach to Optimize Online Dictionary Learning* : Nikhil Rao and Fatih Porikli, IEEE International Conference on Acoustics, Speech and Signal Processing, 2012

13. *Convex Approaches for Group Sparse Signal Recovery in Compressed Sensing* : Workshop on Signal Processing with Adaptive Structured Sparse Representations (SPARS) 2011, Edinburgh, UK

14. *Convex approaches to Model Wavelet Sparsity Patterns* : Nikhil Rao, Robert Nowak, Stephen Wright and Nick Kingsbury, IEEE international Conference on Image Processing, 2011 (1st prize, Best Student Paper Award)

15. *Using Machines to Improve Human Saliency Detection* : Nikhil Rao, Joseph Harrison, Tyler Karrels, Robert Nowak and Timothy Rogers , Asilomar Conference on Signals, Systems and Computers, 2010

Appendix A

Proofs of Theorems

A.1 Proof of Theorem 2.2.5

To prove this result, we need two lemmas:

Lemma A.1.1. *Let $q_1, \ldots, q_L$ be L, χ-squared random variables with d-degrees of freedom. Then*

$$\mathbb{E}[\max_{1\leq i\leq L} q_i] \leq (\sqrt{2\log(L)} + \sqrt{d})^2.$$

Proof. Let $M_L := \max_{1\leq i\leq L} q_i$. For $t > 0$, we have that

$$\begin{aligned}
\mathbb{E}[M_L] &= \frac{\log[\exp(t\cdot \mathbb{E}[M_L])]}{t} \\
&\overset{(i)}{\leq} \frac{\log[\mathbb{E}[\exp(t\cdot M_L)]]}{t} \\
&\overset{(ii)}{=} \frac{\log[\mathbb{E}[\max_{1\leq j\leq L}\exp(t\cdot q_j)]]}{t} \\
&\overset{(iii)}{\leq} \frac{\log[L\mathbb{E}[\exp(t\cdot q_1)]]}{t} \\
&= \frac{\log(L) - \frac{d}{2}\log(1-2t)}{t}
\end{aligned}$$

Where (i) follows from Jensen's inequality , (ii) follows from the monotonicity of the exponential function, and (iii) merely bounds the maximum by the sum over all the elements. Now, setting $t = (2+2\epsilon)^{-1}$ with $\epsilon = \sqrt{\frac{d}{2\log(L)}}$ yields $\mathbb{E}[M_L] \leq (\sqrt{2\log(L)} + \sqrt{d})^2$ □

Note that t can be optimized depending on the application. We use this particular choice because it makes no assumptions about the relative magnitudes of $(M-k)$ and B.

Lemma A.1.2. *Suppose $v \in \mathbb{R}^p$ is supported on some set of groups $\mathcal{G}^\star \subset \mathcal{G}$. Then,*

$$\|v\| \leq \sqrt{|\mathcal{G}^\star|}\ \ \|v\|_{\mathcal{A}}^* .$$

Proof. By duality, it suffices to show that $\|z\|_{\mathcal{A}} \leq \sqrt{|\mathcal{G}^\star|}\ \ \|z\|$ for all z with $supp(z) \subset \mathcal{G}^\star$. For any such z, there exists a representation $z = \sum_{G \in \mathcal{G}^\star} b_G$ where none of the supports of b_G overlap. It then follows that

$$\begin{aligned}
\|z\|_{\mathcal{A}} &\overset{(i)}{\leq} \sum_{G \in \mathcal{G}^\star} \|b_G\| \\
&\overset{(ii)}{\leq} \sqrt{|\mathcal{G}^\star|} \left(\sum_{G \in \mathcal{G}^\star} \|b_G\|^2 \right)^{1/2} \\
&= \sqrt{|\mathcal{G}^\star|}\ \ \|z\|
\end{aligned}$$

Where (i) follows from the definition of the norm $\|\cdot\|_{\mathcal{A}}$ and (ii) is a consequence of the relation $\|\beta\|_1 \leq \sqrt{k}\|\beta\|_2$ for k dimensional vectors β □

Proof of Theorem 2.2.5. Intuition: Note that, from (2.9),the Gaussian width of the intersection of the tangent cone at $x^\star$ with the unit sphere is bounded above by the expected euclidean distance between a random Gaussian vector and the normal cone at $x^\star$ (2.10). We can further bound this distance by the distance between a random Gaussian vector g and a particular vector $r \in \mathcal{N}_{\mathcal{A}}(x^\star)$, shown in (A.1). We proceed to construct such a vector r and prove the result

$$\mathbb{E}_g[\text{dist}(g, C^*)^2] \leq \mathbb{E}_g[\text{dist}(g, r)^2], \ \ r \in \mathcal{N}_{\mathcal{A}}(x^\star) \tag{A.1}$$

Now, let $S = \cup_{G \in \mathcal{G}^\star} G$, *i.e.* S is the indices corresponding to the union of groups that support $x^\star$. Note that $S \subset \{1, 2, \ldots, p\}$.

Since the normal cone is nonempty, there exists a $v \in \mathcal{N}_{\mathcal{A}}(x^\star)$ with $\|v\|_{\mathcal{A}}^* = 1$ and $v_{S^c} = 0$. Since v is in the normal cone, it will also satisfy $\langle v, x^\star \rangle = \|x^\star\|_{\mathcal{A}}$. We will use this v in our bound below.

Let $w \sim \mathcal{N}(0, I_p)$ be a vector with i.i.d. Gaussian entries. We can write $w = [w_S \;\; w_{S^c}]^T$. Let $t(w) = \max_{G \notin \mathcal{G}^\star} \|w_G\|$.

Let us now construct a vector $r \in \mathcal{N}_{\mathcal{A}}(x^\star)$. We can decompose r as $r = [r_S \;\; r_{S^c}]^T$. Let $r_S = t(w) \cdot v_S$, and $r_{S^c} = w_{S^c}$.

From (2.10), and from our definition of $t(w)$, we have $r \in \mathcal{N}_{\mathcal{A}}(x^\star)$. Referring to (2.9), we now consider the expected squared distance between $\mathcal{N}_{\mathcal{A}}(x^\star)$ and w:

$$
\begin{aligned}
\mathbb{E}[\text{dist}(w, C^*)] &\leq \mathbb{E}[||r - w||^2] \\
&\overset{(i)}{=} \mathbb{E}[\|r_S - w_S\|^2 + \|r_{S^c} - w_{S^c}\|^2] \\
&= \mathbb{E}[\|r_S - w_S\|^2] \\
&\overset{(ii)}{=} \mathbb{E}[\|r_S\|^2] + \mathbb{E}[\|w_S\|^2] \\
&= \mathbb{E}[\|t(w) \cdot v_S\|^2] + \mathbb{E}[\|w_S\|^2] \\
&\overset{(iii)}{=} \mathbb{E}[t(w)^2] \cdot \|v_S\|^2 + \mathbb{E}[\|w_S\|^2] \\
&\overset{(iv)}{=} \mathbb{E}[t(w)^2] \cdot \|v_S\|^2 + |S| \\
&\overset{(v)}{\leq} (\sqrt{2\log(M-k)} + \sqrt{B})^2 \cdot \|v_S\|^2 + kB \\
&\overset{(vi)}{\leq} (\sqrt{2\log(M-k)} + \sqrt{B})^2 \cdot k + kB
\end{aligned}
$$

Where (i) follows because S and S^c are disjoint, (ii) follows from the fact that r_S and

w_S are independent, (iii) follows from the fact that v is deterministic. We obtain (iv) since $\|w_S\|^2$ is a χ^2 random variable with $|S|$ degrees of freedom. (v) follows from Lemma A.1.1, and from the fact that kB is a upper bound on the signal sparsity. Finally, (vi) follows from Lemma A.1.2, noting that $|\mathcal{G}^\star| \leq k$, and $\|v\|_{\mathcal{A}}^* = 1$. □

A.2 Proof of Theorem 2.3.3

Here, we look to bound the mean width of the set

$$\mathcal{C}_{sos} = \left\{ \boldsymbol{x} : \|\boldsymbol{x}\| \leq 1, \;\; h(\boldsymbol{x}) \leq \sqrt{kB}(1+\sqrt{\alpha}) \right\}$$

Proof. Before we compute the mean width, let us first compute an expression for the maximum inner product between a random Gaussian vector $\boldsymbol{g} \sim \mathcal{N}(0, \boldsymbol{I})$ and an element in $\mathcal{C}$

$$\begin{aligned}
&\sup \boldsymbol{g}^T \boldsymbol{x} \;\; \textbf{s.t.} \;\; \boldsymbol{x} \in \mathcal{C} \\
&= \sup \sum_{G \in \mathcal{G}} \boldsymbol{g}_G^T \boldsymbol{w}_G \;\; \textbf{s.t.} \;\; \sum_{G \in \mathcal{G}} \left\{ \sqrt{B} \|\boldsymbol{w}_G\|_2 + \|\boldsymbol{w}_G\|_1 \right\} \leq \sqrt{kB}(1+\sqrt{\alpha}) \\
&\overset{(i)}{\leq} \sup \sum_{G \in \mathcal{G}} \boldsymbol{g}_G^T \boldsymbol{w}_G \;\; \textbf{s.t.} \;\; (\sqrt{B}+1) \sum_{G \in \mathcal{G}} \|\boldsymbol{w}_G\|_2 \leq \sqrt{kB}(1+\sqrt{\alpha}) \\
&= \frac{\sqrt{kB}(1+\sqrt{\alpha})}{(\sqrt{B}+1)} \sup \sum_{G \in \mathcal{G}} \boldsymbol{g}_G^T \boldsymbol{w}_G \;\; \textbf{s.t.} \;\; \sum_{G \in \mathcal{G}} \|\boldsymbol{w}_G\|_2 \leq 1 \\
&= \frac{\sqrt{kB}(1+\sqrt{\alpha})}{(\sqrt{B}+1)} \max_{G \in \mathcal{G}} \|\boldsymbol{g}_G\| \qquad \text{(A.2)}
\end{aligned}$$

where (i) follows since the constraint set in (i) is a superset of the constraint set in the expression above it.

Now,

$$\begin{aligned}
\omega(\mathcal{C})^2 &= \left(\mathbb{E}\left[\sup_{\boldsymbol{x}\in\mathcal{C}} \boldsymbol{g}^T\boldsymbol{x}\right]\right)^2 \\
&\overset{(ii)}{\leq} \mathbb{E}\left[\sup_{\boldsymbol{x}\in\mathcal{C}} \boldsymbol{g}^T\boldsymbol{x}\right]^2 \\
&\overset{(iii)}{\leq} \mathbb{E}\left[\frac{\sqrt{kB}(1+\sqrt{\alpha})}{(\sqrt{B}+1)} \max_{G\in\mathcal{G}} \|\boldsymbol{g}_G\|\right]^2 \\
&= \left(\frac{\sqrt{kB}(1+\sqrt{\alpha})}{(\sqrt{B}+1)}\right)^2 \mathbb{E}\left[\max_{G\in\mathcal{G}} \|\boldsymbol{g}_G\|\right]^2 \\
&\overset{(iv)}{=} \left(\frac{\sqrt{kB}(1+\sqrt{\alpha})}{(\sqrt{B}+1)}\right)^2 \mathbb{E}\left[\max_{G\in\mathcal{G}} \|\boldsymbol{g}_G\|^2\right] \\
&\leq k\left(1+\sqrt{\alpha}\right)^2 \mathbb{E}\left[\max_{G\in\mathcal{G}} \|\boldsymbol{g}_G\|^2\right] \qquad \text{(A.3)}
\end{aligned}$$

(ii) follows from Jensen's inequality, (iii) from (A.2). (iv) is a consequence of noting that the square of maximum of non negative quantities is the same as the maximum of the square of the same quantities.

Now, since $\boldsymbol{g} \sim \mathcal{N}(0, \boldsymbol{I})$, $\|\boldsymbol{g}_G\|^2 \sim \chi^2_B$, a chi-squared random variable with at most B degrees of freedom. From Lemma A.1.1, we have

$$\omega(\mathcal{C})^2 \leq k\,(1+\alpha)\left(\sqrt{2\log(M)} + \sqrt{B}\right)^2$$

□

A.3 Proof of Corollary 2.3.5

Before we prove this result, we make note of the following lemma

Lemma A.3.1. *Suppose $\boldsymbol{A} \in \mathbb{R}^{s\times t}$, and let $\boldsymbol{A}_G \in \mathbb{R}^{|G|\times t}$ be the sub matrix of $\boldsymbol{A}$ formed by retaining the rows indexed by group $G \in \mathcal{G}$. Suppose $\sigma_{max}(\boldsymbol{A})$ is the maximum singular value of $\boldsymbol{A}$, and similarly for $\boldsymbol{A}_G$. Then*

$$\sigma_{max}(\boldsymbol{A}) \geq \sigma_{max}(\boldsymbol{A}_G) \;\; \forall G \in \mathcal{G}$$

Proof. Consider an arbitrary vector $\boldsymbol{x} \in \mathbb{R}^p$, and let $\bar{G}$ be the indices that are to indexed by G. We then have the following:

$$\begin{aligned} \|\boldsymbol{A}\boldsymbol{x}\|^2 &= \left\| \begin{bmatrix} \boldsymbol{A}_G\boldsymbol{x} \\ \\ \boldsymbol{A}_{\bar{G}}\boldsymbol{x} \end{bmatrix} \right\|^2 \\ &= \|\boldsymbol{A}_G\boldsymbol{x}\|^2 + \|\boldsymbol{A}_{\bar{G}}\boldsymbol{x}\|^2 \\ \Rightarrow \|\boldsymbol{A}\boldsymbol{x}\|^2 &\geq \|\boldsymbol{A}_G\boldsymbol{x}\|^2 \end{aligned} \tag{A.4}$$

We therefore have

$$\begin{aligned} \sigma_{max}(\boldsymbol{A}) &= \sup_{\|\boldsymbol{x}\|=1} \|\boldsymbol{A}\boldsymbol{x}\| \\ &\geq \sup_{\|\boldsymbol{x}\|=1} \|\boldsymbol{A}_G\boldsymbol{x}\| \\ &= \sigma_{max}(A_G) \end{aligned}$$

where the inequality follows from (A.4). □

We now proceed to prove Corollary 2.3.5.

Proof. Since the entries of the data matrices are correlated Gaussians, the inner products in

the objective function of the optimization problem (2.16) can be written as

$$\langle \boldsymbol{\Phi}_i, \boldsymbol{x} \rangle = \langle \boldsymbol{\Sigma}^{\frac{1}{2}} \boldsymbol{\Phi}'_i, \boldsymbol{x} \rangle = \langle \boldsymbol{\Phi}'_i, \boldsymbol{\Sigma}^{\frac{1}{2}} \boldsymbol{x} \rangle$$

where $\boldsymbol{\Phi}'_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. Hence, we can replace $\boldsymbol{x}$ in our result in Theorem 2.3.4 by $\boldsymbol{\Sigma}^{\frac{1}{2}} \boldsymbol{x}$, and make appropriate changes to the constraint set.

We then see that the optimization problem we need to solve is

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} - \sum_{i=1}^{n} \boldsymbol{y}_i \langle \boldsymbol{\Phi}'_i, \boldsymbol{\Sigma}^{\frac{1}{2}} \boldsymbol{x} \rangle \;\; \textbf{s.t.} \;\; \boldsymbol{x} \in \mathcal{C}_{corr}$$

Defining $\boldsymbol{z} = \boldsymbol{\Sigma}^{\frac{1}{2}} \boldsymbol{x}$, we can equivalently write the above optimization as

$$\hat{\boldsymbol{z}} = \arg\min - \sum_{i=1}^{n} \boldsymbol{y}_i \langle \boldsymbol{\Phi}'_i, \boldsymbol{z} \rangle \;\; \textbf{s.t.} \;\; \boldsymbol{z} \in \boldsymbol{\Sigma}^{\frac{1}{2}} \mathcal{C}_{corr} \tag{A.5}$$

where we define $\Sigma^{\frac{1}{2}} \mathcal{C}$ to be the set $\mathcal{C}$, with each element multiplied by $\boldsymbol{\Sigma}^{\frac{1}{2}}$. We see that (A.5) is of the same form as (2.16), with the constraint set "scaled" by the matrix $\boldsymbol{\Sigma}^{\frac{1}{2}}$. We now need to bound the mean width of the set $\boldsymbol{\Sigma}^{\frac{1}{2}} \mathcal{C}_{corr}$. We then have

$$\begin{aligned}
\max_{\boldsymbol{z}\in\boldsymbol{\Sigma}^{\frac{1}{2}}\mathcal{C}} \boldsymbol{g}^T\boldsymbol{z} &= \max_{\boldsymbol{x}\in\mathcal{C}} \boldsymbol{g}^T\boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{x} \\
&= \max_{\boldsymbol{x}\in\mathcal{C}_{corr}} (\boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{g})^T\boldsymbol{x} \\
&\overset{(i)}{\leq} \max_{\{\boldsymbol{w}_G\}\in\mathcal{W}(\boldsymbol{x})} \sum_{G\in\mathcal{G}} (\boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{g})^T\boldsymbol{w}_G \ \textbf{s.t.} \sum_{G\in\mathcal{G}} \|\boldsymbol{w}_G\|_2 \leq \frac{\sqrt{kB}(1+\sqrt{\alpha})}{\sigma_{min}(\boldsymbol{\Sigma}^{\frac{1}{2}})(1+\sqrt{B})} \\
&= \frac{\sqrt{kB}(1+\sqrt{\alpha})}{\sigma_{min}(\boldsymbol{\Sigma}^{\frac{1}{2}})(1+\sqrt{B})} \max_{G\in\mathcal{G}} \|[\boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{g}]_G\| \\
&\leq \frac{\sqrt{k}(1+\sqrt{\alpha})}{\sigma_{min}(\boldsymbol{\Sigma}^{\frac{1}{2}})} \max_{G\in\mathcal{G}} \|\boldsymbol{\Sigma}_G^{\frac{1}{2}}\boldsymbol{g}\|
\end{aligned}$$

where (i) follows from the same arguments used to obtain (i) in the proof of Theorem (2.3.4). By $\boldsymbol{\Sigma}_G^{\frac{1}{2}}$, we mean the $|G| \times p$ sub matrix of $\boldsymbol{\Sigma}^{\frac{1}{2}}$ obtained by retaining rows indexed by group G.

To compute the mean width, we need to find $\mathbb{E}[\max_{G\in G} \|\boldsymbol{\Sigma}_G^{\frac{1}{2}}\boldsymbol{g}\|^2]$. Now, since $\boldsymbol{g} \sim \mathcal{N}(0, \boldsymbol{I})$, $\boldsymbol{\Sigma}_G^{\frac{1}{2}}\boldsymbol{g} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_G^{\frac{1}{2}}(\boldsymbol{\Sigma}_G^{\frac{1}{2}})^T)$. Hence, $\|\boldsymbol{\Sigma}_G^{\frac{1}{2}}\boldsymbol{g}\|^2 \leq \sigma_{max}(\boldsymbol{\Sigma}_G^{\frac{1}{2}})\|\boldsymbol{c}\|^2$ where $\boldsymbol{c} \sim \mathcal{N}(0, \boldsymbol{I}_{|G|})$. $\|\boldsymbol{c}\|^2 \sim \chi^2_{|G|}$, and we can again use Lemma A.1.1 to obtain the following bound for the mean width:

$$\begin{aligned}
\omega(\boldsymbol{\Sigma}^{\frac{1}{2}}\mathcal{C})^2 &\leq \frac{k(1+\sqrt{\alpha})^2}{\sigma_{min}(\boldsymbol{\Sigma})}(\sqrt{2\log(M)} + \sqrt{B})^2 \left[\max_{G\in\mathcal{G}} \sigma_{\max}(\boldsymbol{\Sigma}_G)\right] \\
&\leq \sigma_{max}(\boldsymbol{\Sigma}) \frac{k(1+\sqrt{\alpha})^2}{\sigma_{min}(\boldsymbol{\Sigma})}(\sqrt{2\log(M)} + \sqrt{B})^2 \qquad \text{(A.6)}
\end{aligned}$$

where the last inequality follows from Lemma A.3.1.

We then have that so long as the number of measurements n is larger than $C\delta^{-2}$ times the

quantity in (A.6),

$$\|\hat{\boldsymbol{z}} - \boldsymbol{z}^\star\|^2 = \left\|\boldsymbol{\Sigma}^{\frac{1}{2}}\hat{\boldsymbol{x}} - \boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{x}^\star\right\|^2 \leq \delta$$

However, note that

$$\sigma_{min}(\boldsymbol{\Sigma})\|\hat{\boldsymbol{x}} - \boldsymbol{x}^\star\|^2 \leq \left\|\boldsymbol{\Sigma}^{\frac{1}{2}}\hat{\boldsymbol{x}} - \boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{x}^\star\right\|^2 \tag{A.7}$$

(A.6) and (A.7) combine to give the final result. Note that for the sake of keeping the exposition simple, we have used Lemma A.3.1 and bounded the number of measurements needed as a function of the maximum singular value of $\boldsymbol{\Sigma}$. However, the number of measurements actually needed only depends on $\max_{G\in\mathcal{G}} \sigma_{max}(\boldsymbol{\Sigma}_G)$, which is typically much lesser.

□

A.4 Proof of Theorem 3.3.2

Theorem 3.3.2 is (except for a minor difference in the upper bounds on η) a true generalization of Theorem 3.3.1, in that we recover the statement of Theorem 3.3.1 by setting $\omega = 0$ in Theorem 3.3.2. Likewise, the *proof* of Theorem 3.3.1 can be obtained by setting $\omega = 0$ in Theorem 3.3.2, so we prove only the latter result here.

Denote $f_t := f(\boldsymbol{x}_t)$, $\tilde{f}_t := f(\tilde{\boldsymbol{x}}_t)$, and $f_t^{FW} := f(\boldsymbol{x}_{t-1} + \gamma_t(\tau\boldsymbol{a}_t - \boldsymbol{x}_{t-1}))$. We have from the algorithm description that

$$f_{t+1} \leq \eta f_t + (1-\eta) f_{t+1}^{FW}.$$

For $\gamma \in [0,1]$, we define

$$\boldsymbol{x}_t(\gamma) := (1-\gamma)\boldsymbol{x}_t + \gamma\tau\boldsymbol{a}_{t+1}.$$

Because Step 6 of Algorithm 1 chooses the value of γ optimally, we have $f_{t+1}^{FW} = f(\boldsymbol{x}_t(\gamma_{t+1})) \leq f(\boldsymbol{x}_t(\gamma))$, for all $\gamma \in [0,1]$, and so

$$\begin{aligned}
& f_{t+1} \\
& \leq \eta f_t + (1-\eta) f_{t+1}^{FW} \\
& \leq \eta f_t + (1-\eta) f(\boldsymbol{x}_t(\gamma)) \\
& \leq \eta f_t + \\
& (1-\eta)\left[f_t + \nabla f(\boldsymbol{x}_t)^T(\boldsymbol{x}_t(\gamma) - \boldsymbol{x}_t)\right] + \\
& (1-\eta)\left[\frac{L}{2}\|\boldsymbol{x}_t(\gamma) - \boldsymbol{x}_t\|^2\right] \quad \text{(by definition of } L\text{)} \\
& = f_t + \\
& (1-\eta)\left[\nabla f(\boldsymbol{x}_t)^T\left((1-\gamma)\boldsymbol{x}_t + \gamma\tau\boldsymbol{a}_{t+1} - \boldsymbol{x}_t\right)\right] + \\
& (1-\eta)\left[\frac{L}{2}\|(1-\gamma)\boldsymbol{x}_t + \gamma\tau\boldsymbol{a}_{t+1} - \boldsymbol{x}_t\|^2\right] \\
& = f_t + (1-\eta)\left[\gamma\nabla f(\boldsymbol{x}_t)^T\left(\tau\boldsymbol{a}_{t+1} - \boldsymbol{x}_t\right)\right] + \\
& (1-\eta)\left[\frac{L\gamma^2}{2}\|\tau\boldsymbol{a}_{t+1} - \boldsymbol{x}_t\|^2\right] \\
& \leq f_t + (1-\eta)\left[\gamma(1-\omega)\nabla f(\boldsymbol{x}_t)^T(\boldsymbol{x}^\star - \boldsymbol{x}_t)\right] + \\
& (1-\eta)\left[2\gamma^2 L R^2\tau^2\right] \quad \text{(see below)} \\
& \leq f_t + (1-\eta)\left[\gamma(1-\omega)(f_\star - f_t) + 2\gamma^2 L R^2\tau^2\right]. \qquad \text{(A.8)}
\end{aligned}$$

The last inequality follows from convexity of the objective function. The second-last inequality uses two results. First, note that the solution $\boldsymbol{x}^*$ can be expressed as follows:

$$\boldsymbol{x}^* = \sum_{\boldsymbol{a}\in\mathcal{A}} c^*_{\boldsymbol{a}}\boldsymbol{a}, \quad \text{for } c^* \geq 0 \text{ with } \textstyle\sum_{\boldsymbol{a}\in\mathcal{A}} c^*_{\boldsymbol{a}} \leq \tau.$$

We therefore have

$$\begin{aligned}
&\langle \nabla f(\boldsymbol{x}_t), \boldsymbol{x}^* - \boldsymbol{x}_t \rangle \\
&= \left\langle \nabla f(\boldsymbol{x}_t), \left(\sum_{\boldsymbol{a}\in\mathcal{A}} c^*_{\boldsymbol{a}}\boldsymbol{a} \right) - \boldsymbol{x}_t \right\rangle \\
&\geq \left(\sum_{\boldsymbol{a}\in\mathcal{A}} c^*_{\boldsymbol{a}} \right) \min_{\boldsymbol{a}\in\mathcal{A}} \langle \nabla f(\boldsymbol{x}_t), \boldsymbol{a} \rangle - \langle \nabla f(\boldsymbol{x}_t), \boldsymbol{x}_t \rangle \\
&\geq \min_{\boldsymbol{a}\in\mathcal{A}} \langle \nabla f(\boldsymbol{x}_t), \tau\boldsymbol{a} - \boldsymbol{x}_t \rangle \\
&\geq \frac{1}{1-\omega} \langle \nabla f(\boldsymbol{x}_t), \tau\boldsymbol{a}_{t+1} - \boldsymbol{x}_t \rangle,
\end{aligned}$$

by the definition of $\boldsymbol{a}_{t+1}$ in (3.11) and noting that $\min_{\boldsymbol{a}\in\mathcal{A}} \langle \nabla f(\boldsymbol{x}_t)\boldsymbol{a} \rangle \leq 0$. Second, we use the definition of R together with $\|\boldsymbol{x}_t\|_{\mathcal{A}} \leq \tau$ and $\boldsymbol{a}_{t+1} \in \mathcal{A}$ to deduce

$$\|\tau\boldsymbol{a}_{t+1} - \boldsymbol{x}_t\| \leq \tau \left(\|\boldsymbol{a}_{t+1}\| + \|\boldsymbol{x}_t/\tau\| \right) \leq 2\tau R,$$

which we can use to bound the squared-norm term. By subtracting f^* from both sides of (A.8), and defining

$$\delta_t := f(\boldsymbol{x}_t) - f^*, \tag{A.9}$$

we obtain that

$$\delta_{t+1} \leq \left[1 - \gamma(1-\eta)(1-\omega) \right] \delta_t + 2(1-\eta) L R^2 \gamma^2 \tau^2, \tag{A.10}$$

for all $\gamma \in [0,1]$. This inequality implies immediately that $\{\delta_t\}_{t=0,1,2,\ldots}$ is a decreasing sequence, since $\gamma = 0$ is always a valid choice in (A.10).

Note that $\delta_0 = f_0 - f_\star = D$. For the first iteration $t = 0$, set $\gamma = 1$ in (A.10) to obtain a further bound on δ_1:

$$\delta_1 \leq [\eta + \omega(1-\eta)]D + 2(1-\eta)LR^2\tau^2 = \tilde{C}_1.$$

For subsequent iterations $t \geq 1$, we consider the following choice of γ:

$$\tilde{\gamma}_t := \frac{\delta_t}{2\tilde{C}_1}.$$

By monotonicity of $\{\delta_t\}$ and the bound above on δ_1, we have $\tilde{\gamma}_t \leq 1/2$ for all $t \geq 1$. By substituting the choice $\gamma = \tilde{\gamma}_t$ into (A.10), we obtain

$$\begin{aligned} \delta_{t+1} &\leq \delta_t - \delta_t^2 \frac{(1-\eta)(1-\omega)\tilde{C}_1 - (1-\eta)LR^2\tau^2}{2\tilde{C}_1^2} \\ &= \delta_t - \frac{\delta_t^2}{\tilde{C}}. \end{aligned} \tag{A.11}$$

The denominator of $\tilde{C}$ is positive because $\eta \in (0, 1/3]$ and $\omega \in (0, 1/4]$ together imply that

$$(1-\omega)\tilde{C}_1 - LR^2\tau^2 > 2(1-\omega)(1-\eta)LR^2\tau^2 - LR^2\tau^2 \geq 0.$$

Note too that

$$\tilde{C} = \frac{2\tilde{C}_1^2}{(1-\eta)((1-\omega)\tilde{C}_1 - LR^2\tau^2)} > 2\tilde{C}_1,$$

so that $\delta_1 \leq \tilde{C}/2$. An argument from [6, Lemma 2.1] yields the result. Since $\delta_1 \leq \tilde{C}/2$,

the bound (3.12) holds for $t = 1$. Since $\{\delta_t\}$ is a decreasing sequence, we have $\delta_t \leq \tilde{C}/2$ for all $t \geq 1$. For the inductive step, assume that (3.12) holds for some $t \geq 1$. Since the right-hand side of (A.11) is an increasing function of δ_t for all $\delta_t \in (0, \tilde{C}/2)$, this quantity can be upper-bounded by substituting the upper bound $\tilde{C}/(t+1)$ for δ_t, to obtain

$$\begin{aligned}\delta_{t+1} &\leq \delta_t - \frac{\delta_t^2}{\tilde{C}} \leq \frac{\tilde{C}}{(t+1)} - \frac{\tilde{C}}{(t+1)^2} \\ &= \frac{\tilde{C}t}{(t+1)^2} = \frac{\tilde{C}t(t+2)}{(t+1)^2(t+2)} \leq \frac{\tilde{C}}{t+2},\end{aligned}$$

establishing the inductive step and completing the proof.

A.5 Proof of Theorem 3.3.3

Here we prove linear convergence for CoGEnT, under the assumptions made in Theorem 3.3.3

We start with a technical lemma.

Lemma A.5.1. *We have*

$$\|\boldsymbol{r}_t\|\|\boldsymbol{w}_t\| \geq \frac{\delta}{\sqrt{\|(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}\|}}\|\boldsymbol{r}_t\|,$$

where δ is defined in (3.15).

Proof. It follows from (3.14) and the Cauchy-Schwatrz inequality that

$$\|\boldsymbol{r}_t\|\|\boldsymbol{w}_t\| \geq |\langle \boldsymbol{r}_t, \boldsymbol{w}_t \rangle| \geq \frac{\delta}{\sqrt{\|(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}\|}}\|\boldsymbol{r}_t\|,$$

giving the result. □

We now prove Theorem 3.3.3.

Proof. Denote $f_t := f(\boldsymbol{x}_t)$, $\tilde{f}_t := f(\tilde{\boldsymbol{x}}_t)$, and $f_t^{FW} := f(\boldsymbol{x}_{t-1} + \gamma_t(\tau \boldsymbol{a}_t - \boldsymbol{x}_{t-1}))$. We have from the algorithm description that

$$f_t \leq \eta f_{t-1} + (1-\eta) f_t^{FW}, \tag{A.12}$$

which we express explicitly as follows:

$$\|\boldsymbol{r}_{t+1}\|^2 = f_t \leq \eta \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}_{t-1}\|^2 + (1-\eta)\frac{1}{2}\|\boldsymbol{y} - \boldsymbol{\Phi}(\boldsymbol{x}_{t-1} + \gamma_t(\tau \boldsymbol{a}_t - \boldsymbol{x}_{t-1}))\|^2.$$

By using definitions of $\boldsymbol{r}_t$ and $\boldsymbol{w}_t$, we obtain

$$\begin{aligned}
\|\boldsymbol{r}_{t+1}\|^2 &\leq \eta\|\boldsymbol{r}_t\|^2 + (1-\eta)\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}_{t-1} - \gamma_t\boldsymbol{\Phi}\tau\boldsymbol{a}_t + \gamma_t\boldsymbol{\Phi}\boldsymbol{x}_{t-1}\|^2 \\
&= \eta\|\boldsymbol{r}_t\|^2 + (1-\eta)\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}_{t-1} - \gamma_t\boldsymbol{\Phi}\tau\boldsymbol{a}_t + \gamma_t\boldsymbol{\Phi}\boldsymbol{x}_{t-1} + \gamma_t\boldsymbol{y} - \gamma_t\boldsymbol{y}\|^2 \\
&= \eta\|\boldsymbol{r}_t\|^2 + (1-\eta)\|(1-\gamma_t)\boldsymbol{r}_t + \gamma_t\boldsymbol{w}_t\|^2 \\
&= \eta\|\boldsymbol{r}_t\|^2 + (1-\eta)\{\|\boldsymbol{r}_t\|^2 + \gamma_t^2\|\boldsymbol{w}_t - \boldsymbol{r}_t\|^2 + 2\gamma_t\langle\boldsymbol{r}_t, \boldsymbol{w}_t - \boldsymbol{r}_t\rangle\} \\
&= \|\boldsymbol{r}_t\|^2 + (1-\eta)\{\gamma_t^2\|\boldsymbol{w}_t - \boldsymbol{r}_t\|^2 + 2\gamma_t\langle\boldsymbol{r}_t, \boldsymbol{w}_t - \boldsymbol{r}_t\rangle\}.
\end{aligned}$$

By substituting the value of γ_t from (3.18) in this expression, we obtain

$$
\begin{aligned}
\|\boldsymbol{r}_{t+1}\|^2 &\leq \|\boldsymbol{r}_t\|^2 + (1-\eta)\left\{-\frac{\langle \boldsymbol{r}_t, \boldsymbol{r}_t - \boldsymbol{w}_t\rangle^2}{\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2}\right\} \\
&= \frac{\|\boldsymbol{r}_t\|^2\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2 - (1-\eta)\langle \boldsymbol{r}_t, \boldsymbol{r}_t - \boldsymbol{w}_t\rangle^2}{\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2} \\
&= \frac{\eta\langle \boldsymbol{r}_t, \boldsymbol{r}_t\rangle^2 - 2\eta\langle \boldsymbol{r}_t, \boldsymbol{r}_t\rangle\langle \boldsymbol{r}_t, \boldsymbol{w}_t\rangle + \langle \boldsymbol{r}_t, \boldsymbol{r}_t\rangle\langle \boldsymbol{w}_t, \boldsymbol{w}_t\rangle - (1-\eta)\langle \boldsymbol{r}_t, \boldsymbol{w}_t\rangle^2}{\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2} \\
&= \frac{\eta\langle \boldsymbol{r}_t, \boldsymbol{r}_t\rangle[\langle \boldsymbol{r}_t, \boldsymbol{r}_t\rangle - 2\langle \boldsymbol{r}_t, \boldsymbol{w}_t\rangle] + \|\boldsymbol{r}_t\|^2\|\boldsymbol{w}_t\|^2 - (1-\eta)\langle \boldsymbol{r}_t, \boldsymbol{w}_t\rangle^2}{\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2} \\
&= \frac{\eta\|\boldsymbol{r}_t\|^2[\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2 - \|\boldsymbol{w}_t\|^2] + \|\boldsymbol{r}_t\|^2\|\boldsymbol{w}_t\|^2 - (1-\eta)\langle \boldsymbol{r}_t, \boldsymbol{w}_t\rangle^2}{\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2} \\
&= \frac{\|\boldsymbol{r}_t\|^2[(1-\eta)\|\boldsymbol{w}_t\|^2 + \eta\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2] - (1-\eta)\langle \boldsymbol{r}_t, \boldsymbol{w}_t\rangle^2}{\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2} \\
&= \frac{(1-\eta)[\|\boldsymbol{r}_t\|^2\|\boldsymbol{w}_t\|^2 - \langle \boldsymbol{r}_t, \boldsymbol{w}_t\rangle^2] + \eta\|\boldsymbol{r}_t\|^2\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2}{\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2} \qquad = \frac{(1} \\
&\leq \frac{(1-\eta)\|\boldsymbol{r}_t\|^2[\|\boldsymbol{w}_t\|^2 - \frac{\delta^2}{\|(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}\|}]}{\|\boldsymbol{r}_t - \boldsymbol{w}_t\|^2} + \eta\|\boldsymbol{r}_t\|^2 \quad \text{by (3.14)} \\
&\leq (1-\eta)\|\boldsymbol{r}_t\|^2\frac{\|\boldsymbol{w}_t\|^2 - \frac{\delta^2}{\|(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}\|}}{\|\boldsymbol{w}_t\|^2} + \eta\|\boldsymbol{r}_t\|^2 \quad \text{by Lemma A.5.1 and (3.16a)} \\
&= \|\boldsymbol{r}_t\|^2\left\{(1-\eta)\left[1 - \left(\frac{\delta}{\sqrt{\|(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}\|}\|\boldsymbol{w}_t\|}\right)^2\right] + \eta\right\} \\
&\leq \|\boldsymbol{r}_t\|^2\left\{(1-\eta)\left[1 - \left(\frac{\delta}{\sqrt{\|(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}\|}[\|\boldsymbol{y}\| + R\tau\|\boldsymbol{\Phi}\|]}\right)^2\right] + \eta\right\} \quad \text{by definition of } \boldsymbol{w}_t.
\end{aligned}
\tag{A.13}
$$

Using the definition of C in the statement of the theorem, we can simplify the quantity inside the braces in (A.13) as follows:

$$(1-\eta)[1-C] + \eta = 1 - (1-\eta)C.$$

By combining this expression with (A.13), we obtain

$$\|\boldsymbol{r}_{t+1}\|^2 \leq [1 - (1 - \eta)C] \|\boldsymbol{r}_t\|^2$$

From Lemma 2.1 in [6], we then have

$$\|\boldsymbol{r}_T\|^2 \leq \|\boldsymbol{r}_0\|^2 \exp(-TC(1 - \eta)),$$

completing the proof. □

List of Figures

List of Tables

List of Algorithms

Bibliography

[1] A. ANANDKUMAR, R. GE, D. HSU, S. M. KAKADE, AND M. TELGARSKY, *Tensor decompositions for learning latent variable models*, preprint arXiv:1210.7559, 2012.

[2] F. BACH, *Consistency of the group lasso and multiple kernel learning*, Journal of Machine Learning Research, 9 (2008), pp. 1179–1225.

[3] F. BACH, *Self-concordant analysis for logistic regression*, Electronic Journal of Statistics, 4 (2010), pp. 384–414.

[4] ——, *Adaptivity of averaged stochastic gradient descent to local strong convexity for logistic regression*, arXiv preprint arXiv:1303.6149, (2013).

[5] R. G. BARANIUK, V. CEVHER, M. F. DUARTE, AND C. HEGDE, *Model-based compressive sensing*, IEEE Transactions on Information Theory, (2010).

[6] A. BECK AND M. TEBOULLE, *A conditional gradient method with linear rate of convergence for solving convex linear systems*, Mathematical Methods of Operations Research, 59 (2004), pp. 235–247.

[7] ——, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM Journal on Imaging Sciences, 2 (2009), pp. 183–202.

[8] P. BICKEL, Y. RITOV, AND A. TSYBAKOV, *Simultaneous analysis of lasso and dantzig selector*, Annals of Statistics, 37 (2009), pp. 1705–1732.

[9] H. D. BONDELL AND B. J. REICH, *Simultaneous regression shrinkage, variable selection, and supervised clustering of predictors with oscar*, Biometrics, 64 (2008), pp. 115–123.

[10] F. BUNEA, *Honest variable selection in linear and logistic regression models via ℓ_1 and $\ell_1 + \ell_2$ penalization*, Electronic Journal of Statistics, 2 (2008), pp. 1153–1194.

[11] J.-F. CAI, E. J. CANDÈS, AND Z. SHEN, *A singular value thresholding algorithm for matrix completion*, SIAM Journal on Optimization, 20 (2010), pp. 1956–1982.

[12] E. J. CANDÈS AND B. RECHT, *Exact matrix completion via convex optimization*, Foundations of Computational Mathematics, 9 (2009), pp. 717–772.

[13] E. J. CANDES, J. ROMBERG, AND T. TAO, *Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information*, IEEE Trans. Information Theory, 52 (2006), pp. 489–509.

[14] V. CHANDRASEKARAN, B. RECHT, P. A. PARRILO, AND A. WILLSKY, *The convex geometry of linear inverse problems*, preprint arXiv:1012.0621v1, (2010).

[15] S. CHATTERJEE, A. BANERJEE, S. CHATTERJEE, AND A. R. GANGULY, *Sparse group lasso for regression on land climate variables.*, in ICDM Workshops, 2011, pp. 1–8.

[16] S. CHEN, D. DONOHO, AND M. SAUNDERS, *Atomic decomposition by basis pursuit*, SIAM Review, 43 (2009), pp. 129– 159.

[17] S. S. CHEN, D. L. DONOHO, AND M. A. SAUNDERS, *Atomic decomposition by basis pursuit*, SIAM J. Scientific Computing, 20 (1998), pp. 33–61.

[18] M. CHERAGHCHI, A. KARBASI, S. MOHAJER, AND V. SALIGRAMA, *Graph-constrained group testing*, in 2010 IEEE International Symposium on Information Theory Proceedings (ISIT), IEEE, 2010, pp. 1913–1917.

[19] M. S. CROUSE, R. D. NOWAK, AND R. G. BARANIUK, *Wavelet based statistical signal processing using hidden markov models.*, Transactions on Signal Processing, 46 (1998), pp. 886–902.

[20] W. DAI AND O. MILENKOVIC, *Subspace pursuit for compressive sensing signal reconstruction*, IEEE Transactions on Information Theory, 55 (2009), pp. 2230–2249.

[21] ——, *Set: an algorithm for consistent matrix completion*, in Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on, IEEE, 2010, pp. 3646–3649.

[22] S. DASGUPTA, D. HSU, AND N. VERMA, *A concentration theorem for projections*, arXiv preprint arXiv:1206.6813, (2012).

[23] D. DONOHO, A. MALEKI, AND A. MONTANARI, *Message passing algorithms for compressed sensing*, National Academy of Sciences, (2009).

[24] D. L. DONOHO, *Compressed sensing*, IEEE Trans. Information Theory, 52 (2006), pp. 1289–1306.

[25] M. F. DUARTE, V. CEVHER, AND R. G. BARANIUK, *Model-based compressive sensing for signal ensembles*, Allerton, (2009).

[26] M. F. DUARTE, M. B. WAKIN, AND R. G. BARANIUK, *Wavelet-domain compressive*

signal reconstruction using a hidden markov tree model, Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), (2008), pp. 5137–5140.

[27] M. DUDIK, Z. HARCHAOUI, AND J. MALICK, *Learning with matrix gauge regularizers*, NIPS Optimization Workshop, (2011).

[28] B. DUMITRESCU, *Positive Trigonometric Polynomials and Signal Processing Applications*, Springer, 2007.

[29] J. C. DUNN, *Rates of convergence for conditional gradient algorithms near singular and nonsingular extremals*, SIAM Journal on Control and Optimization, 17 (1979), pp. 187–211.

[30] E. FEREDOES, G. TONONI, AND B. R. POSTLE, *The neural bases of the short-term storage of verbal information are anatomically variable across individuals*, The Journal of Neuroscience, 27 (2007), pp. 11003–11008.

[31] M. FRANK AND P. WOLFE, *An algorithm for quadratic programming*, Naval Research Logistics Quarterly, 3 (1956), pp. 95–110.

[32] R. M. FREUND AND P. GRIGAS, *New analysis and results for the conditional gradient method*, preprint arXiv:1307.0873, 2013.

[33] Y. GORDON, *On Milman's inequality and random subspaces which escape through a mesh in* $\mathbb{R}^n$, Geometric aspects of functional analysis, Isr. Semin., 1317 (1986 - 87), pp. 84–106.

[34] J. GUELAT AND P. MARCOTTE, *Some comments on wolfe's Ôaway stepÕ*, Mathematical Programming, 35 (1986), pp. 110–119.

[35] J. HUANG AND T. ZHANG, *The benefit of group sparsity*, Technical report, arXiv:0901.2962. Preprint available at http://arxiv.org/pdf/0903.2962v2, (2009).

[36] J. HUANG, T. ZHANG, AND D. METAXAS, *Learning with structured sparsity*, Technical report, arXiv:0903.3002. Preprint available at http://arxiv.org/pdf/0903.3002v2, (2009).

[37] L. JACOB, G. OBOZINSKI, AND J. P. VERT, *Group lasso with overlap and graph lasso*, Proceedings of the 26th International Conference on machine Learning, (2009).

[38] M. JAGGI, *Revisiting Frank-Wolfe: Projection-free sparse convex optimization*, in Proceedings of the 30th International Conference on Machine Learning, 2013, pp. 427–435.

[39] P. JAIN, A. TEWARI, AND I. S. DHILLON, *Orthogonal matching pursuit with replacement*, Advances in Neural Information Processing Systems, (2011), pp. 1215–1223.

[40] A. JALALI, P. D. RAVIKUMAR, S. SANGHAVI, AND C. RUAN, *A dirty model for multi-task learning.*, 3 (2010), p. 7.

[41] R. JENATTON, J. AUDIBERT, AND F. BACH, *Structured variable selection with sparsity inducing norms*, Technical report, arXiv:0904.3523. Preprint available at http://arxiv.org/pdf/0904.3523v3, (2009).

[42] R. JENATTON, J. MAIRAL, G. OBOZINSKI, , AND F. BACH, *Proximal methods for hierarchical sparse coding*, Technical report, arXiv:1009.3139. submitted, (2010).

[43] R. JENATTON, J. MAIRAL, G. OBOZINSKI, AND F. BACH, *Proximal methods for*

hierarchical sparse coding, The Journal of Machine Learning Research, 12 (2011), pp. 2297–2334.

[44] C. C. JOHNSON, A. JALALI, AND P. D. RAVIKUMAR, *High-dimensional sparse inverse covariance estimation using greedy methods*, International Conference on Artificial Intelligence and Statistics, (2012), pp. 574–582.

[45] R. H. KESHAVAN AND S. OH, *A gradient descent algorithm on the grassman manifold for matrix completion*, arXiv preprint arXiv:0910.5260, (2009).

[46] C. LA AND M. N. DO, *Tree based orthogonal matching pursuit algorithm for signal reconstruction*, IEEE International Conference on Image Processing, Atlanta, GA., (2006), pp. 1277 – 1280.

[47] S. LACOSTE-JULIEN, M. JAGGI, M. SCHMIDT, P. PLETSCHER, ET AL., *Block-coordinate Frank-Wolfe optimization for structural SVMs*, International Conference on Machine Learning, (2013), pp. 53–61.

[48] J. LIU, R. FUJIMAKI, AND J. YE, *Forward-backward greedy algorithms for general convex smooth functions over a cardinality constraint*, preprint arXiv:1401.0086, 2013.

[49] K. LOUNICI, M. PONTIL, A. B. TSYBAKOV, AND S. VAN DE GEER, *Taking advantage of sparsity in multi-task learning*, arXiv preprint arXiv:0903.1468, (2009).

[50] J. MAIRAL, R. JENATTON, G. OBOZINSKI, AND F. BACH, *Convex and network flow optimization for structured sparsity*, The Journal of Machine Learning Research, 12 (2011), pp. 2681–2720.

[51] A. MAURER AND M. PONTIL, *Structured sparsity and generalization*, The Journal of Machine Learning Research, 13 (2012), pp. 671–690.

[52] M. B. MCCOY, V. CEVHER, Q. T. DINH, A. ASAEI, AND L. BALDASSARRE, *Convexity in source separation: Models, geometry, and algorithms*, preprint arXiv:1311.0258, 2013.

[53] L. MEIER, S. VAN DE GEER, AND P. BÜHLMANN, *The group lasso for logistic regression*, Journal of the Royal Statistical Society: Series B (Statistical Methodology), 70 (2008), pp. 53–71.

[54] S. MENDELSON, A. PAJOR, AND N. TOMCZAK-JAEGERMANN, *Reconstruction and subgaussian operators in asymptotic geometric analysis*, Geometric and Functional Analysis, 17 (2006), pp. 1248 – 1282.

[55] M. MISHALI AND Y. ELDAR, *Blind multi-band signal reconstruction: compressed sensing for analog signals*, IEEE Trans. Signal Processing, 57 (2009), pp. 993–1009.

[56] S. MOSCI, S. VILLA, A. VERRI, AND L. ROSASCO, *A primal-dual algorithm for group sparse regularization with overlapping groups*, Neural Information Processing Systems, (2010).

[57] S. MOSCI, S. VILLA, A. VERRI, AND L. ROSASCO, *A primal-dual algorithm for group sparse regularization with overlapping groups*, in Advances in Neural Information Processing Systems, 2010, pp. 2604–2612.

[58] D. NEEDELL AND J. TROPP, *Cosamp: Iterative signal recovery from incomplete and inaccurate samples*, Appl. Comput. Harmon. Anal., 26 (2008), pp. 301–321.

[59] S. NEGAHBAN, P. RAVIKUMAR, M. WAINWRIGHT, AND B. YU, *A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers.*, Preprint ArXiv :1010:2731v1, (2010).

[60] S. NEGAHBAN AND M. J. WAINWRIGHT, *Joint support recovery under high-dimensional scaling: Benefits and perils of l1-regularization*, Advances in Neural Information Processing Systems, 21 (2008), pp. 1161–1168.

[61] Y. NESTEROV, *Gradient methods for minimizing composite objective functions*, Mathematical Programming, Series B, (2013). To appear.

[62] J. NOCEDAL AND S. WRIGHT, *Numerical Optimization*, Springer, second ed., 2006.

[63] G. OBOZINSKI, L. JACOB, AND J.-P. VERT, *Group lasso with overlaps: the latent group lasso approach*, arXiv preprint arXiv:1110.0413, (2011).

[64] D. PERCIVAL, *Theoretical properties of the overlapping groups lasso*, Preprint arXiv:1103.4614v2 [stat.ML], (2011).

[65] Y. PLAN AND R. VERSHYNIN, *Robust 1-bit compressed sensing and sparse logistic regression: A convex programming approach*, Information Theory, IEEE Transactions on, 59 (2013), pp. 482–494.

[66] N. RAO, C. COX, R. NOWAK, AND T. T. ROGERS, *Sparse overlapping sets lasso for multitask learning and its application to fmri analysis*, (2013), pp. 2202–2210.

[67] N. RAO, R. NOWAK, S. WRIGHT, AND N. KINGSBURY, *Convex approaches to model wavelet sparsity patterns*, IEEE International Conference on Image Processing, (2011).

[68] N. S. RAO, B. RECHT, AND R. D. NOWAK, *Universal measurement bounds for structured sparse signal recovery*, in International Conference on Artificial Intelligence and Statistics, 2012, pp. 942–950.

[69] G. RASKUTTI, M. J. WAINWRIGHT, AND B. YU, *Restricted eigenvalue properties for correlated gaussian designs*, The Journal of Machine Learning Research, 11 (2010), pp. 2241–2259.

[70] B. RECHT, *A simpler approach to matrix completion*, Journal of Machine Learning Research, 12 (2011), pp. 3413–3430.

[71] I. RISH, G. A. CECCHIA, K. HEUTONB, M. N. BALIKIC, AND A. V. APKARIANC, *Sparse regression analysis of task-relevant information distribution in the brain*, in Proceedings of SPIE, vol. 8314, 2012, p. 831412.

[72] T. ROCKAFELLAR AND J. B. WETS, *Variational analysis*, Springer Series of Comprehensive Studies in Mathematics, 317 (1997).

[73] J. K. ROMBERG, H. CHOI, AND R. G. BARANIUK, *Bayesian tree structured image modeling using wavelet domain hidden markov models*, Transactions on Image Processing, (2000).

[74] S. RYALI, K. SUPEKAR, D. A. ABRAMS, AND V. MENON, *Sparse logistic regression for whole brain classification of fmri data*, NeuroImage, 51 (2010), p. 752.

[75] P. SCHNITER, *Turbo reconstruction of structured sparse signals*, Proc. Conference on Information Sciences and Systems, (2010).

[76] N. SIMON, J. FRIEDMAN, T. HASTIE, AND R. TIBSHIRANI, *A sparse-group lasso*, Journal of Computational and Graphical Statistics, 22 (2013), pp. 231–245.

[77] S. SOM AND P. SCHNITER, *Compressive imaging using approximate message passing and a markov-tree prior*, IEEE transactions on signal processing, (2011).

[78] P. SPRECHMANN, I. RAMIREZ, G. SAPIRO, AND Y. ELDAR, *Collaborative hierarchical sparse modeling*, in Information Sciences and Systems (CISS), 2010 44th Annual Conference on, IEEE, 2010, pp. 1–6.

[79] P. SPRECHMANN, I. RAMIREZ, G. SAPIRO, AND Y. C. ELDAR, *C-hilasso: A collaborative hierarchical sparse modeling framework*, Signal Processing, IEEE Transactions on, 59 (2011), pp. 4183–4198.

[80] A. SUBRAMANIAN ET AL., *Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression proÞles*, National Academy of Sciences, 102 (2005), p. 15545Ð15550.

[81] G. TANG, B. N. BHASKAR, P. SHAH, AND B. RECHT, *Compressive sensing off the grid*, in 50th Annual Allerton Conference on Communication, Control, and Computing, IEEE, 2012, pp. 778–785.

[82] A. TEWARI, P. K. RAVIKUMAR, AND I. S. DHILLON, *Greedy algorithms for structurally constrained high dimensional problems*, in Advances in Neural Information Processing Systems, 2011, pp. 882–890.

[83] R. TIBSHIRANI, *Regression shrinkage and selection via the lasso*, Journal of the Royal Statistical Society. Series B, (1996), pp. 267–288.

[84] J. A. TROPP AND A. C. GILBERT, *Signal recovery from random measurements via orthogonal matching pursuit*, IEEE Transactions on Information Theory, 53 (2007), pp. 4655–4666.

[85] B. A. TURLACH, W. N. VENABLES, AND S. J. WRIGHT, *Simultaneous variable selection*, Technometrics, 47 (2005), pp. 349–363.

[86] M. J. VAN DE VIJVER, Y. D. HE, L. J. VAN'T VEER, H. DAI, A. A. HART, D. W. VOSKUIL, G. J. SCHREIBER, J. L. PETERSE, C. ROBERTS, M. J. MARTON, ET AL., *A gene-expression signature as a predictor of survival in breast cancer*, New England Journal of Medicine, 347 (2002), pp. 1999–2009.

[87] M. VAN GERVEN, C. HESSE, O. JENSEN, AND T. HESKES, *Interpreting single trial data using groupwise regularisation*, NeuroImage, 46 (2009), pp. 665–676.

[88] X. WANG, T. M. MITCHELL, AND R. HUTCHINSON, *Using machine learning to detect cognitive states across multiple subjects*, CALD KDD project paper, (2003).

[89] A. E. WATERS, A. C. SANKARANARAYANAN, AND R. G. BARANIUK, *Sparcs: Recovering low-rank and sparse matrices from compressive measurements.*, in NIPS, 2011, pp. 1089–1097.

[90] S. J. WRIGHT, R. D. NOWAK, AND M. A. T. FIGUEIREDO, *Sparse reconstruction by separable approximation*, Transactions on Signal Processing, 57 (2009), pp. 2479–2493.

[91] M. YUAN AND Y. LIN, *Model selection and estimation in regression with grouped variables*, Journal of the royal statistical society. Series B, 68 (2006), pp. 49–67.

[92] X. ZENG AND M. A. T. FIGUEIREDO, *The atomic norm formulation of oscar regularization with application to the conditional gradient algorithm*, (2014).

[93] T. ZHANG, *Adaptive forward-backward greedy algorithm for learning sparse representations*, IEEE Transactions on Information Theory, 57 (2011), pp. 4689–4708.

[94] Y. ZHOU, R. JIN, AND S. HOI, *Exclusive lasso for multi-task feature selection*, in International Conference on Artificial Intelligence and Statistics, 2010, pp. 988–995.

[95] H. ZOU AND T. HASTIE, *Regularization and variable selection via the elastic net*, Journal of the Royal Statistical Society: Series B (Statistical Methodology), 67 (2005), pp. 301–320.

Biography

Nikhil Rao was born in Bombay (now Mumbai), India. He received a Bachelor's degree in Electronics and Communications Engineering from the University of Mumbai in 2008. He received an M.S. degree in Electrical and Computer Engineering at the University of Wisconsin-Madison in 2010, and has since been a PhD student at the same University, advised by Professor Robert Nowak. Nikhil won the Best Student Paper Competition at the IEEE International Conference of Image Processing in 2011. He has served as a reviewer for the IEEE Transactions on Signal Processing, IEEE Transactions on Information Theory, IEEE Transactions on Image Processing, and the Annals of Statistics journals. His current research interests are in high dimensional statistical signal processing and machine learning with a focus on optimization and algorithms.

Starting fall of 2014, Nikhil will be a Postdoctoral Researcher at the University of Texas at Austin, with Professor Inderjit Dhillon, as a recipient of the ICES Postdoctoral Fellowship.